A Patient Agent Controlled Customized Blockchain Based Framework for Internet of Things

July, 2021

Md. Ashraf Uddin

School of
Engineering, IT and Physical Sciences

(Doctor of Philosophy(Information Technology))
Federation University Australia

Dissertation submitted to
School of Engineering, IT and Physical Sciences
of
Federation University Australia
for
partial fulfillment of the requirements for the degree of
Doctor of Philosophy.

Written under the supervision of

Associate Professor Andrew Stranieri

and Associate Supervisor
Professor Iqbal Gondal
and
Dr. Venki Balasubramanian

Federation University Australia, July 2021.

To Whom It May Concern

We hereby certify that this is a typical copy of the original doctor thesis of
Md. Ashraf Uddin

Signature of

the Supervisor

Assoc/Prof Andrew Stranieri

Seal of

School of

Engineering, IT and Physical Sciences

Abstract

Although Blockchain implementations have emerged as revolutionary technologies for various industrial applications including cryptocurrencies, they have not been widely deployed to store data streaming from sensors to remote servers in architectures known as Internet of Things. New Blockchain for the Internet of Things models promise secure solutions for eHealth, smart cities, and other applications.

These models pave the way for continuous monitoring of patient's physiological signs with wearable sensors to augment traditional medical practice without recourse to storing data with a trusted authority. However, existing Blockchain algorithms cannot accommodate the huge volumes, security, and privacy requirements of health data.

In this thesis, our first contribution is an End-to-End secure eHealth architecture that introduces an intelligent Patient Centric Agent. The Patient Centric Agent executing on dedicated hardware manages the storage and access of streams of sensors generated health data, into a customized Blockchain and other less secure repositories. As IoT devices cannot host Blockchain technology due to their limited memory, power, and computational resources, the Patient Centric Agent coordinates and communicates with a private customized Blockchain on behalf of the wearable devices.

While the adoption of a Patient Centric Agent offers solutions for addressing continuous monitoring of patients' health, dealing with storage, data privacy and network security issues, the architecture is vulnerable to Denial of Services(DoS) and single point of failure attacks.

To address this issue, we advance a second contribution; a decentralised eHealth system in which the Patient Centric Agent is replicated at three levels: Sensing Layer, NEAR Processing Layer and FAR Processing Layer. The functionalities of the Patient Centric Agent are customized to manage the tasks of the three levels. Simulations confirm protection of the architecture against DoS attacks.

Few patients require all their health data to be stored in Blockchain repositories but instead need to select an appropriate storage medium for each chunk of data by matching their personal needs and preferences with features of candidate storage mediums. Motivated by this context, we advance third contribution; a recommendation model for health data storage that can accommodate patient preferences and make storage decisions rapidly, in real-time, even with streamed data. The mapping between health data features and characteristics of each repository is learned using machine learning.

The Blockchain's capacity to make transactions and store records without central oversight enables its application for IoT networks outside health such as underwater IoT networks where the unattended nature of the nodes threatens their security and privacy. However, underwater IoT differs from ground IoT as acoustics signals are the communication media leading to high propagation delays, high error rates exacerbated by turbulent water currents. Our fourth contribution is a customized Blockchain leveraged framework with the model of Patient-Centric Agent renamed

as Smart Agent for securely monitoring underwater IoT. Finally, the smart Agent has been investigated in developing an IoT smart home or cities monitoring framework. The key algorithms underpinning to each contribution have been implemented and analysed using simulators.

To My Parents-Amena Begum and Md Kuddus

Acknowledgements

It is my great pleasure to express my heartiest thankfulness to those who gave me the valuable time and supported me in making this dissertation possible. I believe that you are the greatest blessing in my life. Thanks all of you to make my dream successful.

I owe my deepest sense of gratitude to my honorable supervisor, Assoc/Prof Andrew Stranieri for his excellent supervision, meaningful suggestions, persistent encouragements, and other fruitful help during each stage of my Ph.D. study. His thoughtful comments and guidance helped me to complete my research papers and present them in productive ways. Besides, he was always patient and helpful whenever his guidance and assistance were needed in both of my academic and daily life in Australia. I have really been lucky in working with a person like him. Needless to say, it would not have been possible to complete this thesis without his guidance and active support.

I am indebted to my Ph.D. associate supervisors, Professor Iqbal Gondal and Dr. Venki Balasubramanian, for taking the valuable time to give me advice, guidance, insightful comments, and proofreading of this thesis. Professor Iqbal Gondal also managed funds for presenting several international conference papers.

I want to express my profound gratitude to Professor Joarder Kamruzzaman for his valuable suggestions and comments during my candidature confirmation. I would like to take this opportunity to thank Professor Joarder Kamruzzaman for his advice and guidance while applying for the scholarship at Federation University. Needless to say, without Joarder's guidance, I would not have had the opportunity to study PhD in Federation University under an excellent supervision team. I like to thank Sajal Halder who first advised me to contact with Professor Joarder Kamruzzaman.

I would like to acknowledge the school of Engineering, IT and Physical Sciences, Federation University for financially supporting my Ph.D. study, and the Jagannath University, Dhaka, Bangladesh for giving me the study leave permission for this study.

I would like to thank for the fruitful discussions and cooperation with many people including Dr. Alireza Jolfaei (Macquarie University), Dr. Ammar Alazab (Melbourne Institute of Technology). I would like to convey my respect to all the members of Internet Commerce Security Lab including Md. Moniruzzaman, Ms Ansam Khraisat for their support during the period of this study. I especially want to thank my beloved wife Omi Akter, who always comforts, consoles, and encourages me. I also like to thank Dr. Md Manowarul Islam (Assoc/Prof, Jagannath University), for his various kinds of support throughout my PhD study.

Last but not least, I am grateful to my family, my mother, father, brothers, and all of my friends. For your unconditional loves, supports, patience, and confidence in me are the biggest rewards as well as the driving forces of my life.

Md. Ashraf Uddin
Federation University Australia
July 2021

List of Publications

Journal Papers

1. **M. A. Uddin**, A. Stranieri, I. Gondal, V. Balasubramanian, The adoption of Blockchain in IoT: Challenges and Solutions, Blockchain Research and Application, 100006, ELSEVIER, 2021.

2. **M. A. Uddin**, A. Stranieri, I. Gondal and V. Balasubramanian, "Continuous Patient Monitoring With a Patient Centric Agent: A Block Architecture," IEEE Access, IEEE, vol. 6, pp. 32700-32726, 2018. doi: 10.1109/ACCESS.2018.2846779

3. **M. A. Uddin**, A. Stranieri, I. Gondal and V. Balasubramanian, "A patient agent to manage blockchains for remote patient monitoring," Studies in health technology and informatics, IOS Press, vol. 254, pp. 105–115, 2018.

4. **M. A. Uddin**, A. Stranieri, I. Gondal and V. Balasubramanian, " Blockchain Leveraged Decentralized IoT eHealth Framework," Internet of Things, ELSEVIER, vol. 9, pp. 100159, 2020. https://doi.org/10.1016/j.iot.2020.100159.

5. **M. A. Uddin**, A. Stranieri, I. Gondal and V. Balasubramanian, " Rapid Health Data Repository Allocation using Predictive Machine Learning," Health Informatics Journal, SAGE, 2020. https://doi.org/10.1177/1460458220957486.

6. **M. A. Uddin**, A. Stranieri, I. Gondal, V. Balasurbramanian,(2019), "A Lightweight Blockchain Based Framework for Underwater IoT", Electronics, MDPI, 8(12), 1552. doi:10.3390/electronics8121552

International Conference Papers

7. **Md Ashraf Uddin**, Andrew Stranieri, Iqbal Gondal, and Venki Balasubramanian. 2020. " Dynamically Recommending Repositories for Health Data: a Machine Learning Model." In Proceedings of the Australasian Computer Science Week Multiconference (ACSW '20). Association for Computing Machinery, New York, NY, USA, Article 24, 1–10. DOI:https://doi.org/10.1145/3373017.3373041

8. **M. A. Uddin**, A. Stranieri, I. Gondal and V. Balasubramanian, "An Efficient Selective Miner Consensus Protocol in Blockchain Oriented IoT Smart Monitoring," 2019 IEEE International Conference on Industrial Technology (ICIT), Melbourne, Australia, 2019, pp. 1135-1142. doi: 10.1109/ICIT.2019.8754936

9. **M. A. Uddin,** A. Stranieri, I. Gondal and V. Balasubramanian,"A Decentralized Patient Agent Controlled Blockchain for Remote Patient Monitoring," 2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Barcelona, Spain, 2019, pp. 1-8. doi: 10.1109/WiMOB.2019.8923209

10. **M. A. Uddin**, A. Stranieri, I. Gondal and V. Balasubramanian, Blockchain Leveraged Task Migration in Body Area Sensor Networks, 2019 25th Asia-Pacific Conference on Communications (APCC), Ho Chi Minh City, Vietnam, 2019, pp. 177-184.

The Candidate's Contribution in Each Chapter

The chapters of this thesis include different publications that were published throughout the candidate's PhD research. The Table 1 maps the chapters to publications and presents the candidate's contribution in each publication.

Table 1: The candidate's contributions in the publications

Chapter No.	Article Title (that covered the chapter)	Status(Year, Publisher)	Contributions (%)
Chapter 1	Introduction		90%
Chapter 2	A Survey on Adoption of Blockchain in IoT: Challenges and Solutions	Published in Blockchain: Research and Applications, 2021, Elsevier	90%
Chapter 3	Continuous Patient Monitoring With a Patient Centric Agent: A Block Architecture	Published in IEEE Access, 2018, IEEE	80%
Chapter 4	Blockchain Leveraged Decentralized IoT eHealth Framework	Published in Internet of Things, 2020, Elsevier	80%
Chapter 5	Rapid Health Data Repository Allocation using Predictive Machine Learning	Published in Health Informatics Journal, 2020, SAGE	80%
Chapter 6	A Lightweight Blockchain Based Framework for Underwater IoT	Published in Electronics, 2019, MDPI	80%
Chapter 7	An Efficient Selective Miner Consensus Protocol in Blockchain Oriented IoT Smart Monitoring	Published in 2019 IEEE International Conference on Industrial Technology (IEEE ICIT)	85%
Chapter 8	Conclusion		90%

List of Figures

List of Tables

Contents

Chapter 1

Introduction

1.1 Introduction

Nowadays, the Internet of Things (IoT) has attracted interest from academics and entrepreneurs thanks to their tremendous ability to provide innovative services through various applications[1]. IoT seamlessly interconnects heterogeneous devices and objects to create a physical system in which sensing, processing, and communication are automatically controlled without human intervention[2]. With the advent of smart homes, smart cities and other intelligent things, IoT has become a field of immense influence, opportunity and development with Cisco Inc. anticipating 50 billion connected devices by 2020[3]. The Wireless Sensor Networks (WSNs) and Machine-to-Machine (M2M) or Cyber-Physical Systems (CPS) have now emerged in the research literature as indispensable elements for the broader term IoT. Consequently, security concerns relating to WSN, M2M, or CPS arise in IoT with the standard network IP protocol.The entire application infrastructure must be protected against attacks that can obstruct IoT services as well as endanger data protection, privacy or confidentiality.

Blockchain first successfully applied in cryptocurrencies has potentially emerged to be a highly secure and privacy-preserving technology in IoT applications[4, 5]. Blockchain (BC) refers to a decentralized, tamper-proof and transactional database that provides a secure way to store and process information across a large number of network participants[6]. In current settings, large quantities of data produced from large numbers of IoT devices may bottleneck the IoT system, resulting in a poor quality of service (QoS)[7].

A single point of failure refers to a part of a system that can interrupt the whole network from running if it crashes, which is undesirable in any system with a goal of high availability or reliability[8]. The peer-to-peer network of the Blockchain is seen as a possible solution to problems with a single point of failure and bottleneck[9, 10]. The adoption of Blockchain in IoT might be adequate means to securely and efficiently store and process IoT data[6].

Blockchain technology has evolved as an important remedy for eliminating trust in conventional authorities or more broadly, online intermediaries, as BC supposedly removes the need for trust amongst entities. In BC technology, participants are subject to authority of a technological mechanism rather than using authority of a centralized organization that is often perceived to be untrustworthy. Filippi et al.[11] made a point that Blockchain-based systems are intended to create trust in a particular system, not by entirely removing trust, but rather by maximising the degree of confidence between participants as a means of indirectly reducing the need for trust. BC allows a circle of trust between independent parties who do not agree to rely on a single third-party

trust. This confidence or trust can be achieved more readily because of technical arrangements, particularly open-source software which indicates that to the extent, the code of a specific piece of software can be open, the possible outcome can be more readily predicted theoretically. Therefore, the higher predictability of the software code, the greater belief in the system and the lower need for faith in that technical system's developers or operators. For instance, anybody can study the open Bitcoin protocol. As a result, this assures participants that the network will produce a certain amount of new Bitcoins (12.5 bitcoins) at a particular speed(one Block per 10 minutes) when a miner wins in Proof of Work without relying on any financial institution or a centralized authority. Therefore, BC technology makes participants believe that no one needs to be trusted, and none can pretend to be a trusted party, as no one exists in the BC[12].

However, the integration of Blockchain and IoT with the restricted power and storage resources, are challenged by the structure of Blockchain that involves high computational costs and delay[13]. The challenges while handling IoT data on the Blockchain are depicted in Figure 1.1 and summarized below.

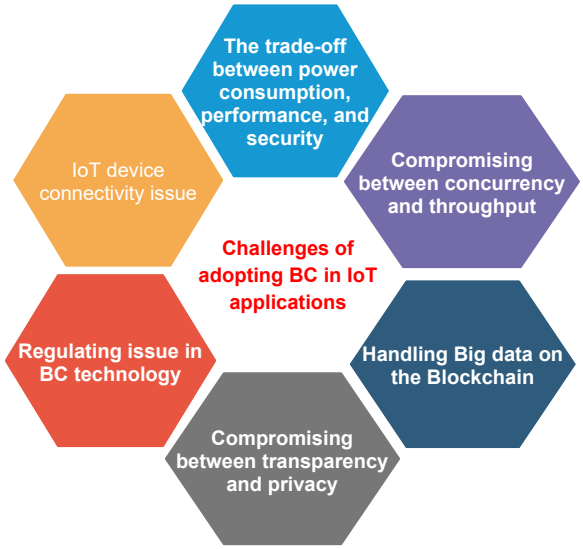


Figure 1.1: The challenges of adopting Blockchain in IoT

- **The trade-off between power consumption, performance, and security:** The high computational power required to run Blockchain algorithms has slowed down the advancement of these technology-based applications on resource constrained devices. For instance, Bitcoin's energy consumption was compared with the domestic power consumption of Ireland, which IoT devices cannot undertake[14]. Zhou et al. [9] reported that the entire Bitcoin network absorbs considerably more energy than several nations, including Austria and Colombia. In addition, researchers have questioned the performance of Blockchain to process IoT data and suggested optimizing its central algorithms to increase the number of confirmed Blocks per second[7]. Elimination of Blockchain Proof of Work (PoW) consensus mechanism can reduce the power consumption and improve performances[15]. But, PoW helps prevents malicious, Sybil attacks and makes the Blocks tamper-proof. Consequently, the goal is to refine Blockchain processes to appropriately align security and efficiency[8].

- **Data concurrency and throughput issue[7]:** In IoT systems, the IoT devices continuously stream data which results in high concurrency[16]. The Blockchain throughput is limited

thanks to its complex cryptographic security protocol and consensus mechanisms. The rapid synchronization of new Blocks among BC nodes in a chain-structured ledger requires a higher amount of bandwidth, which can improve BC throughput[9, 17]. Therefore, the challenge is to boost Blockchain's throughput to meet the need of frequent transactions in IoT systems.

- **Connectivity challenges of IoT[18]:** The IoT devices are expected to be connected to high computing storage and networking resources to share IoT data with potential stakeholders. The IoT has limited capabilities to connect them with BC technology in order to provide novel business opportunities for the implementation of new applications and services in various domains.

- **Handling Big data on the Blockchain:** In the Blockchain network, every participant maintains a local copy of the complete distributed ledger. Upon the confirmation of a new Block, the Block is broadcast throughout the entire peer-to-peer network, and every node appends the confirmed Block to their local ledger. While this decentralised storage structure improves efficiency, solves the bottleneck problem and removes the need for third-party trust[19], the management of IoT data on the Blockchain puts a burden on participants' storage space. The study in [20] calculated that a Blockchain node would need approximately 730 GB of data storage per year if 1000 participants exchange a single 2 MB image per day in a Blockchain application. Therefore, the challenge is to address the increasing data storage requirements when Blockchain deals with IoT data.

- **Challenges in maintaining both transparency and privacy:** Blockchain can guarantee the transparency of the transactions, which is essential in some applications like finance. However, user's confidentiality may be adversely affected when storing and accessing IoT data from certain IoT systems such as eHealth on the BC[21]. To maintain a balanced degree of transparency and privacy, the development of cost-effective access control for IoT using Blockchain is necessary.

- **Regulating challenges of BC in IoT:** While several BC technological features including decentralization, immutability, anonymity, and automation are promising security solutions for diverse IoT applications, these features combinedly pose various new regulatory challenges[22]. The immutability feature implies that data is permanently published in DLT (decentralized ledger technology) on the peer-to-peer network and cannot be deleted or modified. In addition, due to the absence of governance, records cannot be filtered for maintaining privacy before publishing them on the BC. Actions resulting from executing code such as smart contracts on a DLT can breach law. Due to the anonymity of the DLT, it is not so straightforward to distinguish the parties carrying out transactions for illegal services. Whilst the automation feature of the BC brings many advantages, the actors that cause some behaviours including errors in code and obfuscating code are ambiguous. Current IoT laws and regulations are becoming outdated especially with the advent of new disruptive technology such as Blockchain and need to be revised to undertake the DLT[23].

To deal with the challenges mentioned above, this thesis aims to enable low-power, resource-constrained IoT devices to securely transmit and access their data to a customized Blockchain managed storage system.To achieve this, a smart Agent is introduced to connect low-profiled IoT devices with a peer-to-peer Blockchain network where the Agent performs the following important

tasks: 1) decide repositories for permanently storing IoT data, 2) implement access control on the Blockchain and, 3) manage the mining process on behalf of the IoT devices. The software Agent adopts the customized Blockchain to provide users with a decentralized trustless network for processing IoT data, and a distributed storage for selective kinds of IoT data. Furthermore, the Agent makes secure communication between IoT devices and Blockchain, and regulates the Blockchain's consensus mechanism to minimize power consumption and improve the Blockchain's throughput.

The rest of the chapter is organized as follows: the chapter starts with the basics of Blockchain technologies. Section 1.2 provides an overview of Blockchain's fundamental components, the merits and demerits of this technology when applied in IoT applications along with its technical limitations. Our research motivation is described in Section 1.3. Section 1.4 introduces the research goals as well as research questions. The contributions of the thesis are outlined in the section 1.5. Finally, the section 1.6 describes the organization of the rest of the thesis.

1.2 Basics of Blockchain Technology

Blockchain is mostly known as underlying technology of the virtual Bitcoin cryptocurrency invented by Satoshi Nakamoto in 2008. In a nutshell, the Blockchain is typically defined as a transparent, trusted, and decentralised ledger on a peer-to-peer network[8]. The data unit in the Blockchain is called a transaction. A certain number of transactions are packed into a Block. A decentralized Blockchain ledger is created with all confirmed Blocks. A Block in the distributed ledger is linked to the previously approved Block through a cryptographic hash code of the Block [24].This emerging technology has already been widely explored to develop a range of applications beyond digital cryptocurrencies.

Every participant on a peer-to-peer network can verify the behaviour of other participants within the network, as well as make, verify and approve a new transaction to be recorded in the Blockchain. This infrastructure guarantees stable and efficient Blockchain operations with the benefits of tamper resistance and no vulnerabilities to a single point of failure[25]. The Blockchain ledger is available to all participants but still not regulated by any network body. This principle is accomplished by imposing strict rules and mutual agreement between the network nodes, which is characterised as the consensus mechanism. The consensus mechanism refers to the process of synchronising the decentralised ledger across all the nodes in the Blockchain network. Figure 1.2 provides an overview of how Bitcoin Blockchain operates. The key components of a Blockchain network are first discussed in this section. Next we investigate Blockchain's core properties with respect to immutability, security, and integrity. Blockchain comprises several major components that are described below.

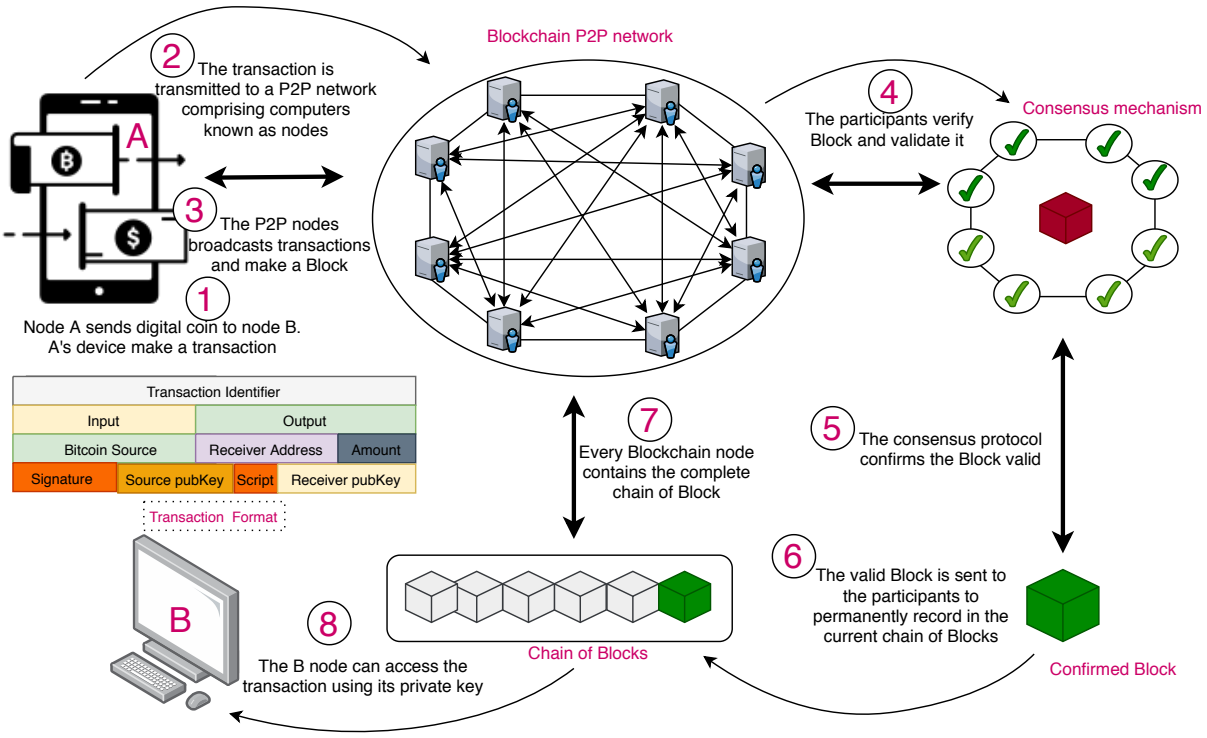


Figure 1.2: The basic operation of a standard Blockchain

- [1]We assume that a participant **A** needs to transfer some digital coins to another participant **B**. **A**'s device makes a transaction as depicted in Figure 1.2. Participants usually can use their portable devices such as smartphone, laptop and low-processing computer for making transactions. The transactions are signed with **A**'s private key and encrypted with the **B**'s public key or symmetric key if it is necessary.

- [2] **A**'s device transmits the transaction to a peer-to-peer network comprising of high-processing devices called nodes. The Blockchain algorithms are implemented on this network.

- [3]The nodes participated in the Blockchain network replicate the transaction and broadcast it throughout the network. The BC nodes packed a certain number of transaction in a Block. The structure of a typical Block is depicted in Figure 1.3. All BC nodes compete to produce a target hash code of the Block which is called Proof of Work.

- [4,5,6] All the participants append the Block to existing chain of confirmed Blocks only if it has been verified. This process is termed as consensus mechanism. Researchers introduced different consensus protocols which varies in terms of computational cost and turnaround time required. Some of widely used consensus mechanisms are discussed in the later section.

- Finally, [8] **B**'s device can access the transaction from the chain of confirmed Blocks using its private key.

1.2.1 Data Block

Blockchain is fundamentally a chain of Blocks, a linear structure that starts with a so-called genesis Block and continues with each new confirmed Block connected to that chain. Each Block comprises several transactions and has a field containing the hash tag of its immediately preceding Block, which creates links between them. Consequently, all confirmed Blocks in the chain can be traced back through cryptographic hash code, and modification or alteration to data of any Block is not possible. A typical data Block is divided into two parts: transaction records and a header. A transaction is made when a user carries out activities on the Blockchain network. For instance, a transaction with associated metadata and signed with a user's private key for ensuring trust is created if the user exchanges money or makes a contract on the Blockchain.

Transaction records are organised in a Merkle tree as depicted in Figure 1.3. A Merkle tree refers to a binary tree structure that summarizes and allows content to be checked efficiently and securely within a large data set. If the transactions are not packed into Merkle trees, each of the network nodes would need to keep a complete copy of each transaction which has ever taken place on the Blockchain[7]. A Merkle tree summarises all transactions within a Block by generating a digital fingerprint of the entire collection of transactions, enabling a user to check whether a transaction is included in a Block or not. If a single transaction is modified or altered, so is the Merkle Root. One field in the Block's header contains the Merkle root generated while making the Block. Merkle trees are generated by hashing node pairs repeatedly until just one hash is left (this hash is called the root hash, or the root of Merkle). Figure 1.3 shows that each leaf node holds a hash of transaction data, and each non-leaf node contains a hash code of its all previous hash code.

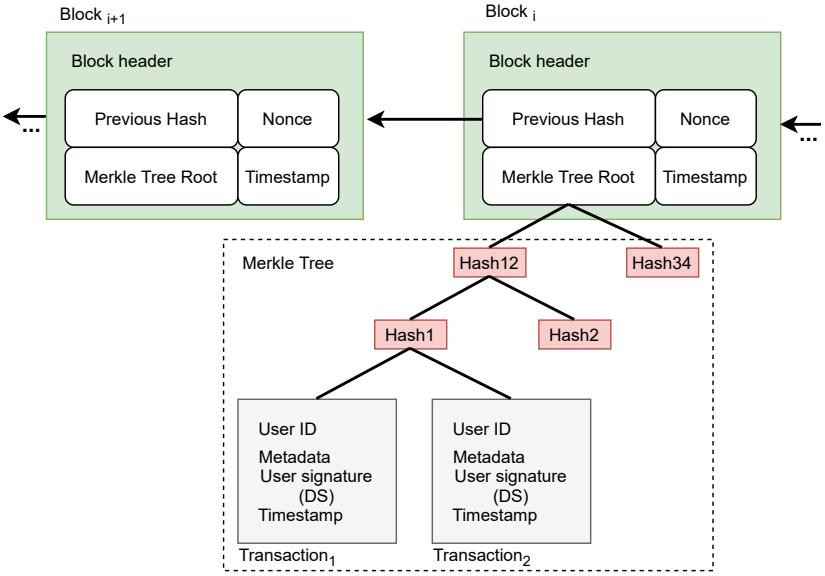


Figure 1.3: The structure of a Block

In general, the Block header includes: 1) a Block hash for authentication, 2) a Merkle tree root that packs a group of transactions to guarantee integrity of data 3) a Nonce that is utilized to produce a hash value below a target level by means of a consensus mechanism and 4) a Timestamp referring to the time the Block has been created. Figure 1.4 adopted from [26] demonstrates a typical procedure of producing digest from the header of a BC Block. The header is partitioned into two portions. The first portion is fed to an SHA 256 hash function which output as Initialize Vector (IV) along with the second portion of the header is input to the second SHA 256 hash

function. Finally, the digest from the second SHA256 hash function with 256 bits padding is fed to the third SHA 256 function to produce the final digest from the Block header. With PoW, the nonce field in the Block header is continuously incremented by Miner nodes until the target hash code is generated.

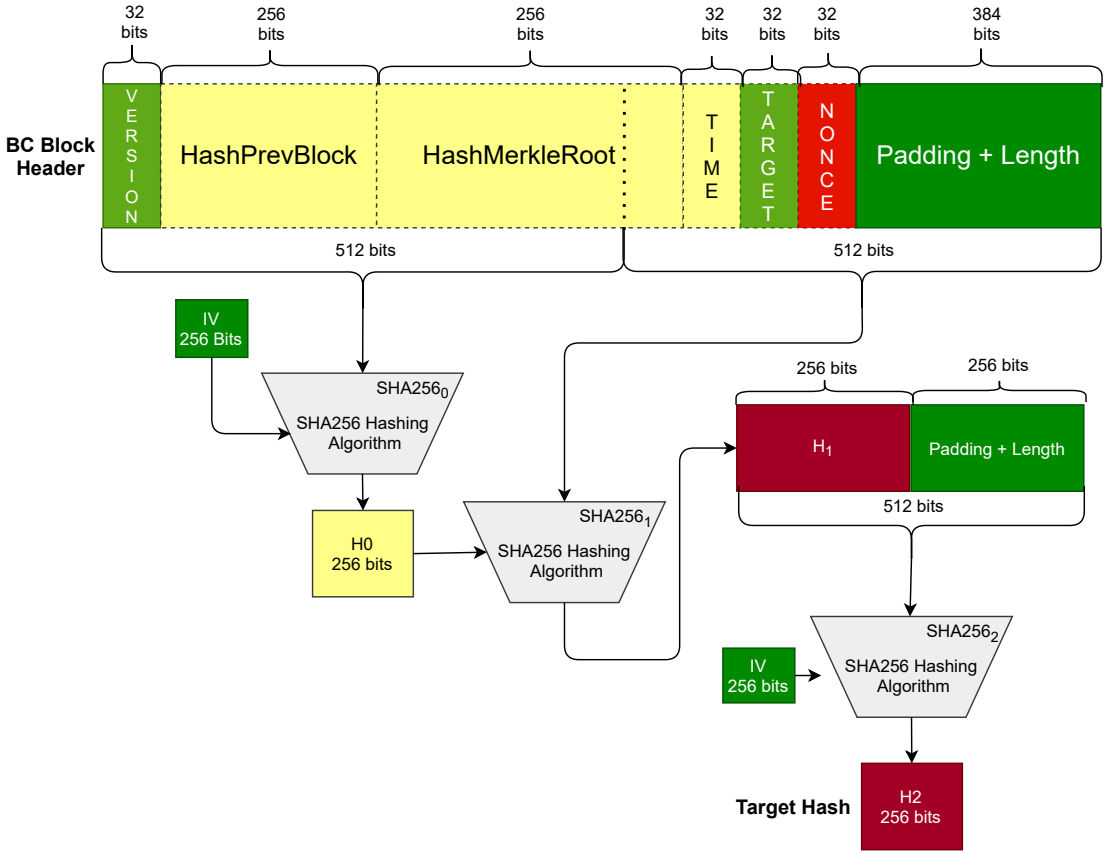


Figure 1.4: The Bitcoin Block Header Hashing Algorithm

The Block is shared between the participants on the peer-to-peer network and each participant links the Block to the existing chain of Block if the Block is approved by the consensus mechanism described in the later section. Thus, a decentralized ledger is formed on the Blockchain and each node stores one copy of that ledger. This eliminates the need of the central control point, which results in high levels of equity between Blockchain participants and providing higher network security. In addition, each Block in the distributed ledger always has a distinct cryptographic signature associated with a timestamp which renders the ledger auditable and unchangeable.

1.2.2 Consensus Protocol

In BC technologies, when a node exchanges a transaction on a P2P (peer to peer) network, no centralised body monitors the transaction or prevent attackers from manipulating or altering data of the transaction. To prevent transactions from fraud related issues such as double-spending attacks, the trustworthiness of a transaction/Block needs to be checked and the data flow should be controlled to ensure smooth exchange of information in the BC network[27]. These requirements are met using validation protocols called as consensus mechanism. In the context of Blockchain, a consensus

mechanism refers to a method of achieving agreement on a single Block between multiple independent nodes. Several consensus standard mechanisms retrieved from state-of-the-art research are described below and presented in Figure 2.4. Five categorizations of consensus mechanisms are shown in Figure 2.4: Proof of Work(PoW), Proof of Stake(PoS), Byzantine Fault Tolerance(BFT), Proof of Authority(PoA) and Proof of Elapsed Time(PoET). Some of these protocols pertaining to our contributions are described below.

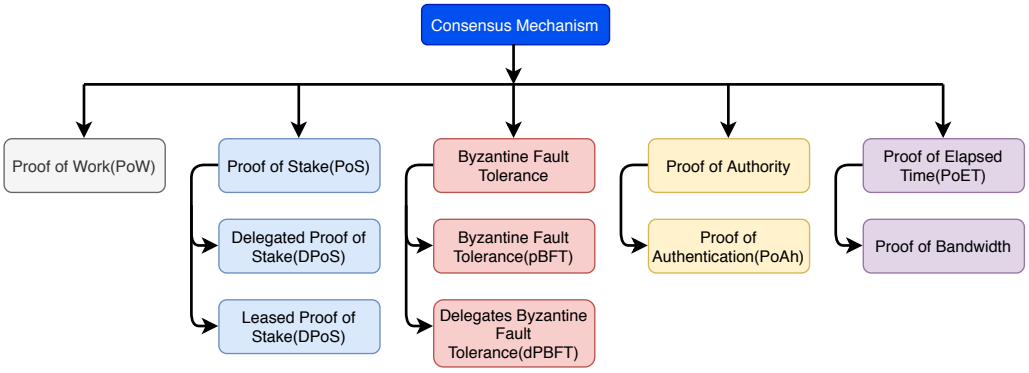


Figure 1.5: The taxonomy of consensus mechanism

- **Practical Byzantine Fault Tolerance(pBFT):** Byzantine Fault Tolerance(BFT)[28] derived from Byzantine general problems refers to reach an agreement between nodes in a distributed network even if some nodes of the network fail to respond or reply with false information. The BFT mechanism can defend network failures through collective decision-making that reduces the impact of the flawed nodes. Practical BFT consensus mechanism is described following ways:

  1. A centralized authority chooses a group of nodes[29]. One node from this group is elected as primary nodes, often called a leader. Other nodes that are also to be chosen as primary node by turn are called secondary nodes or backup nodes.
  2. Next, a client's request is submitted to the primary node.
  3. After that, the primary/leader node broadcasts the request throughout the network so that all the secondary/backup nodes receive the request.
  4. Both primary and secondary nodes perform the service requested by the client and send it a reply. The service is successfully confirmed if the client receives m + 1 number of replies with the same result where m is the number of defective nodes permitted within the network.

  Practical BFT effectively works well in distributed networks with limited number of nodes but with this protocol, communication overhead exponentially increases for each additional node joining in the network. Further, practical BFT is prone to Sybil attacks where one participant maintains many identifiers that can influence BFT mechanism[30].

- **Proof of Elapsed Time (PoET)** [25, 31, 32]: In Proof of Elapsed Time, participants in the Blockchain must wait for a random time. The participant who first finishes the waiting period is nominated as a leader for making new Block. However, a participant can intentionally

choose a short wait time for being a winner. Further, the winner might not complete its waiting time. To address this issue, Intel introduced SGX(Intel Software Guard Extensions) that allows running trusted code for an application in a secure environment. SGX[10] referring to a particular set of CPU instructions prevents participants from manipulating waiting time in PoET. Intel SGX creates a certificate which ensures that a specific trusted code was correctly run in a protected environment. A new participant is required to download the trusted program to join the Blockchain. The trusted program executed on SGX hardware generates an SGX certificate for the participant, which includes the user's public key. The participant sends this certificate to the rest of the network requesting permission to join the Blockchain. The trusted program provides the participant with a timer object authenticated using the private key of the trusted program. The participant is required to wait for the time specified in the timer object. This approach is much more energy-efficient than other consensus protocols such as Proof of Work.

- **Leased Proof of Stake:** Leased Proof of Stake(LPoS)[33] is a variation of the standard Proof of Stake consensus protocol. In regular PoS, every node with a certain amount of digital coin is eligible to mine the next Block. But, nodes that hold a higher amount of cryptocurrency have a higher chance of winning in the mining process. As a result, nodes with low digital coin are unlikely to succeed to mine the next Block or need to wait for long periods. LPoS has been suggested to overcome the drawbacks of the standard PoS. With LPoS, a participant owning a low stake can lease or rent its stake to a full node (a staking node that has higher stake and wants to take part in mining), which increases the full node's chance of becoming the next miner. The leased funds remain under the holder's complete control. If the staking node wins mining to add a Block, it receives incentives that are to be proportionally shared between the staking node and its leasing nodes. A Blockchain user has the choice of operating as a full node or leasing their stake to a full node to earn a proportional reward.

- **Delegated Proof of Stake(DPoS)** [34]: In DPoS, users can vote for the nodes that invest resources in the Blockchain system. The strength of a user's vote is proportional to the number of tokens the user holds[9]. As a result, a group of rich nodes can dominate the network and decide who will be the witness. The nodes with a higher number of votes called the witness are responsible for making Blocks and get paid for their services. Nevertheless, as the network expands, the witness has to compete to remain paid. Voting in this Protocol is an ongoing operation. The network users disqualify a witness if the witness plays bad roles in processing the Block.

1.2.3 Type of Blockchains

Blockchain can be categorized into two major types: public Blockchain (or less permitted) and private Blockchain (or permitted). A public Blockchain is a non-restrictive, permission-less distributed ledger system that allows anyone to join the network and make transactions as well as engage in the process of consensus[35]. Bitcoin and Ethereum with open source and smart contracts are the most prominent public Blockchains. Public Blockchains are mostly reliable if the users strictly abide by the rules and regulations of the Blockchain[36]. On the other hand, private Blockchain is a central authority-operated invitation-only network and a validation process allows participants to validate Blockchain transactions. Another kind of Blockchain is a consor-

tium Blockchain which is a semi-decentralized and governed by a group rather than a single entity. Other kinds of Blockchain are discussed in the next chapter.

1.2.4 Objectives of Blockchain Technology in managing IoT:

The Blockchain maintains a distributed, trusted and tamper proof digital ledger which is stored in thousands of nodes on a peer-to-peer network. The advent of Blockchain technology has brought many benefits across a variety of industries in trustless environments[37]. In this section, a few of the advantages of the Blockchain in IoT are described and presented in Figure 1.6

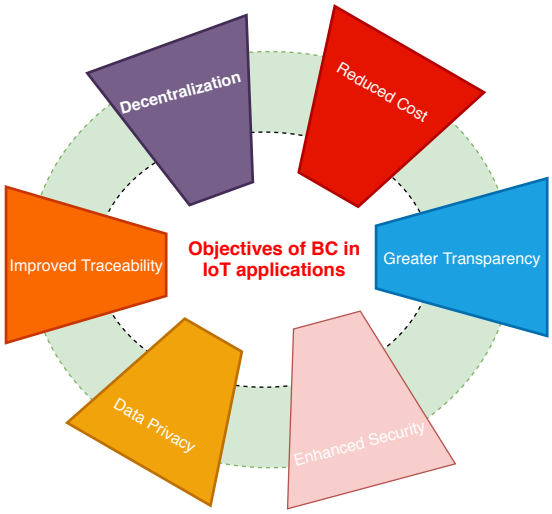


Figure 1.6: The objectives of BC

- **Decentralization:** Blockchain, with its decentralised nature, is a promising technique for effectively solving bottleneck and one-point failure problems by eliminating the need for a trusted third party in the IoT network[6]. The disruption of a Blockchain node does not affect the operation of the Blockchain and IoT network. Blockchain data is usually stored in multiple nodes on the peer-to-peer network. Hence, the system is highly resistant to technological failures and malicious attacks. The availability or security of the network cannot be compromised even if some of the nodes go offline. In contrast, many traditional databases rely on one or more servers. Thus they are more prone to cyber-attacks and technological failure. On the other hand, the peer-to-peer architecture of Blockchain empowers all network attendees with fair validation rights to check IoT data correctness and guarantee immutability.

- **Enhanced Security:** Blockchain is more reliable and secure than other record-keeping systems in several aspects[6]. Transactions must be agreed in advance of being registered. A transaction is encrypted and linked to the previous transaction upon approval of the transaction. In addition to that, information is stored across a network of computers rather than on a single server, which stop hackers from compromising transaction data. In Blockchains, the main element of security is the use of private and public keys. To secure transactions between participants, Blockchain systems use asymmetrical cryptography. These keys are generated with random number and strings so that an individual cannot mathematically create the private key from its public key. This protects Blockchain documents from future

attacks, reduces data leakage problems and strengthens the security of Blockchain network. Blockchain is able to radically change the exchange of confidential information to deter fraud and criminal activity in any field where it is necessary to protect sensitive data from various applications, including financial services, government, and healthcare. Furthermore, with Blockchain-enabled smart contracts[38], the combination of Blockchain and IoT can provide consumers with trusted access control, which automatically authorizes all operations of IoT devices. In addition, smart contract services offer data provenance to users. This enables data owners to control the exchange of their data on Blockchain. Blockchain enables users to define access rules for self-executing smart contracts, which guarantees the privacy and ownership of personal data. Malicious access is verified and disabled through smart contract authorisation and user identification capability.

- **Improved Traceability:** Goods traded in a complex supply chain using a traditional ledger cannot be traced back to their point of origin as quickly in other systems as in Blockchain. Historical data transactions in Blockchain can assist in checking the authenticity of assets and avoiding fraudulent activities. Similarly, the Blockchain can store and track the past records of a patient that are important for patient's care.

- **Greater Transparency:** The transaction histories in Blockchain are more transparent since these histories are available to all network users. Blockchain is a sort of distributed network in which all participants share the same documents as opposed to individual copies in the standard network. This shared document can be modified only by means of a consensus, meaning that everyone has to agree. In other words, the same copy of Blockchain data spreads over a wide network for public verification. Consequently, all Blockchain users have fair rights over the network to link, verify and track transaction activities. To alter a single transaction record, all subsequent records would need to be modified and more than 51% nodes would need to collude in the entire network. As a result, information is more accurate, robust and transparent on the Blockchain than on the conventional network. Such transparency also leads to protecting the credibility of the Blockchain-based systems by reducing the possibility of unauthorised data alternations.

- **Data Privacy:** Thanks to Blockchain's immutable and trustworthy features, storage systems on the Blockchain are an extremely efficient to protect IoT data from alteration[17]. Blockchain archives data transactions and events in an integrity-preserved, authenticity-guaranteed manner by means of immutable hash chains and digital signatures. Essentially, the Blockchain allows users to monitor transactions across the network so that computer and data rights are retained.

- **Reduced Cost:** Cost reduction is one of the main aims for many businesses. Blockchain does not require third parties or middleman and infrastructure's deployment cost for public BC to guarantee business operations, which can reduce the cost of operating business[39]. Blockchain users do not need to review too much paperwork to complete a transaction, as each party has access to a single, and unchangeable ledger. While BC can escape the cost of third-party services, Blockchain requires huge investment for dedicated infrastructure for private and consortium Blockchain and public BC still charges a certain fee for transaction processing (e.g. Gas in Ethereum BC).

- **Immutability:**Transaction on the Blockchain remains immutable over time. Technically, transactions are timestamped after being checked by the Blockchain network and then in-

serted into a Block that is cryptographically protected by a hashing method. Hash mechanism links Blocks together and constructs a sequential chain. One field of a new Block's header always holds the hash value of metadata of the previous Block, which makes the chain strongly immutable[40]. This way, the Block data cannot be updated, altered or removed after it is validated and recorded in the Blockchain. The cryptographic link between subsequent Blocks can withstand any attempts of transactions' alteration or modification. Even if any changes occur in a transaction, it will be easily identified.

1.2.5 Technical Limitations of Blockchain:

While Blockchain is increasingly committed to providing disrupting infrastructure for the Internet of Things, its implementation remains a series of critical challenges to be addressed in terms of scalability, computational cost, security and privacy[32].

- **Scalability issue:**Current Blockchain platforms have bottlenecks in terms of scalability with restricted throughput, low efficiency and higher computational cost. Currently, due to Block size constraints, many Blockchains have lengthy processing periods for transactions to be appended into the chain. Consequently, the Block time increases rapidly, which reduces the overall performance of a system. Furthermore, if all transactions are to be saved on chain, nodes of the blockchain require incredibly higher capacities of storage[41]. Given complex IoT scenarios such as smart cities, IoT data is huge and thus the magnitude of data would grow rapidly which makes the processing of high volumes of data complicated and delay in the Blockchain network. Due to these limitations, many applications developers are not being inspired to build large scale Blockchain IoT systems[42].

- **High computational cost:** Wood et al. [43] reported the cost of completing a transaction as the computational cost of a Blockchain. The processing of a transaction involves various steps, including defining heavy security, mining, validating, and storing it across multiple participants[44]. These steps combinedly consume considerable computing power. There are a variety of mining techniques such as PoW, PoS, pBFT described above that require various levels of energy. For instance, PoW, which is the most decentralized mining process, solves a complicated mathematical puzzle that requires powerful computational hardware to perform transaction validation. Due to resource constraints of IoT systems, it is difficult for them to meet resource requirements of PoW for qualifying the most decentralized nature. Even for IoT devices with fairly large computational capabilities, the sophistication of the Blockchain system will demand heavy technical and human resources. This would trigger consumer's concerns about high running costs that would limit large scale implementation of Blockchain-based systems.

- **Security and privacy threats:** Although Blockchain can withstand major security attacks such as Sybil, DDoS, selfish mining and Ransomware attacks, the existing Blockchain has some inherent security flaws. If more than half of the machines running Blockchain can control computing resources, they can alter consensus processes and stop the confirmation of new transactions for malicious purposes. This is also referred to as a 51% attack in the Bitcoin philosophy. Without robust monitoring of transactions, Blockchain can be at risk of data loss and network disruption. In Sybil attack, the malicious nodes create several identities to either flood the network with transactions or make false statements, such as fake traffic congestion[5, 8]. Distributed service denial (DDoS) attacks are difficult to conduct

on a network of Blockchain. Still, Blockchain technology is susceptible to DDoS attacks and these attacks are actually the most common type on Blockchain networks. DDoS attackers attempt to disrupt the network's mining pools, e-wallets, crypto exchanges and other financial services. Selfish mining is a bitcoin mining strategy in which groups of miners collaborate in order to increase their earnings. A miner (or group of miners) tends to increase their revenue through selfish mining by strategically withholding and releasing Blocks into the network[24]. Although the Blockchain IoT model can support safe data sharing, storing all health data on the Blockchain would slow down transactions, and threaten data leakage and disclosure of security issues for sensitive patient information.

1.3 Research Motivation

In the manufacturing field, IoT technologies have promoted industrial automation and digitization. Various IoT apps recenty developed have improved the quality, flexibility and scalability of manufacturing processes and thus it has reduced error, saved cost, enhanced performance, and security in the manufacturing and industrial process[6]. Most existing IoT architectures maintain a centralized data centre for storing and processing sensors' data, which can be at risk of breaching security, single point failure and malicious attacks like DDoS, Sybil attack [6, 8][45] . This results in unavailability of service and the divulge of sensor data and thus outweigh the important advantages of IoT system. Further, the data interception can occur when IoT devices transfer data between them which questions the reliability of the collected data. The notion of integrating Blockchain and IoT has recently gained significant popularity among the researchers to exploit such hybrid architectures to address the aforementioned issues. However, the adoption of the Blockchain technology into IoT applications poses a couple of challenges outlined in Figure 1.7 such as different mining rate, and imbalanced resources capacity between IoT devices and the Blockchain nodes.

To meet these issues, researchers[13,46–49] suggested autonomous agents to adopt Blockchain technology in various IoT ecosystems including healthcare, smart cities, smart home and electric energy trading which are regulated and managed by the autonomous agents on behalf of the users. An agent typically refers to an autonomous entity which can perform actions on sensors or IoT data on behalf of users. Internet of Things ecosystem compromising a wide variety of devices including wearable sensors, smartphones, network devices and portable computers generate massive quantities of data at very high speed. Users are not always in a position to manage this influx of data[50]. Hence, autonomous entities are required to track and analyse data while streaming the data from different types of IoT devices. The autonomous agent is a proactive body, which can decide the appropriate sensor data actions and automatically trigger action without the human user's intervention[46]. Machine learning and artificial intelligent technology typically form a basis for the creation of an autonomous agent to process and automatically identify action on the data streaming from sensors or online sources[47]. For instance, Tom et al.[46] proposed an agent based smart energy distribution system on the IoT Fog network. The agent is designed to negotiate energy demands with the home agent at the customer's end based on prices and energy availability during peak periods.

The convergence of Blockchain technologies and multi-agents such as agent for environmental protection, energy trading and monitoring patients can handle sensitive data to advocate transparency and trustworthy interactions for consumers and service providers[47]. Luo et al.[49] proposed multi-agents controlled Blockchain based decentralized electricity trading system. This system consists of two layers: upper layer that contains multi-agents for negotiating electricity trading

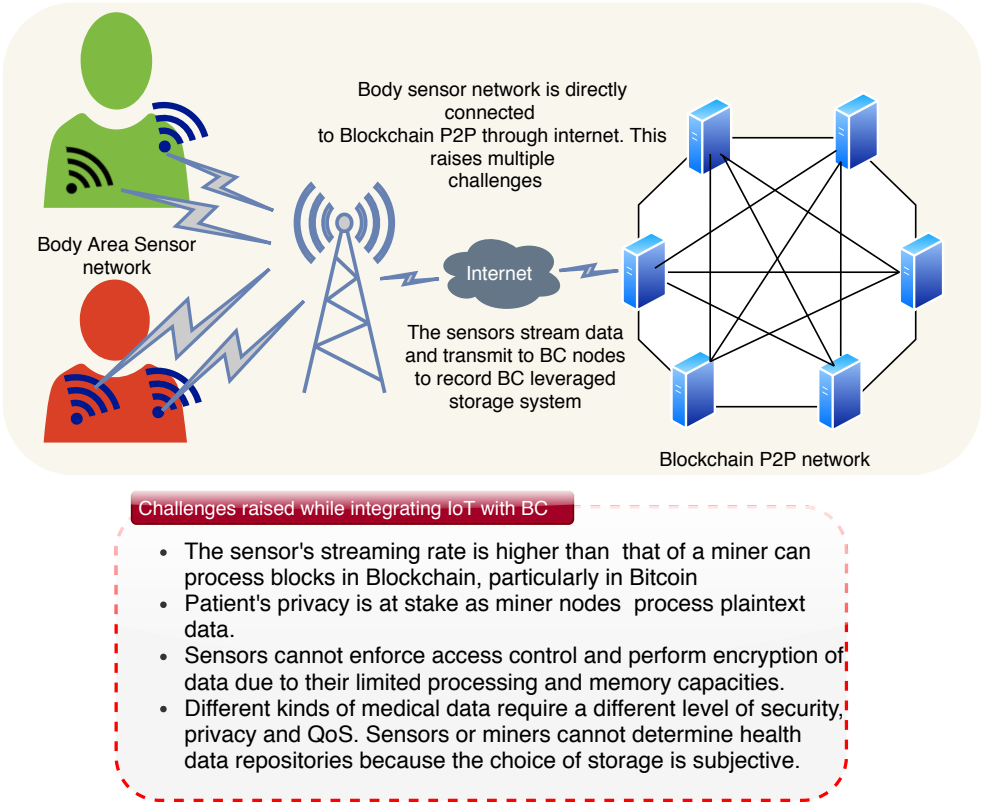


Figure 1.7: The challenges raised to connect Body Area Sensors with Blockchain

contracts, and lower layer that hosts Blockchain network for the settlement of the electricity contracts. Qayumi et al.[48] proposed multi-agents to solve the scalability issue of the Blockchain based architecture but did not describe how this can be achieved. Norta et al. [51] presented smart contracts for the cooperation across various organizations. They described the possibilities of the Blockchain smart contract in realising non-repudiating properties. However, these works are still at a preliminary stage and will be developed in future. With Multi-Agent System (MAS), a software agent working on behalf of IoT devices is an efficient way to promote social interactions between intelligent devices. IoT devices need to associate them with a secure software agent when switching from one area to another[52, 53]. However, IoT devices do not have any accurate information available about the agents in a new environment. Agents are often unknown and not appropriately referenced, and the traditional approach of asking other trusted agents for information is usually impracticable for IoT devices. The research in [54,55] suggested a reputation model of the software agent in which the consumer's feedback for its services is summed up. Ethereum Blockchain[55] was used to preserve and certify the reputation of all the agents in the distributed IoT networks.

Further, researchers[13][56] have attempted to design autonomous algorithms on smart Gateway to adopt Blockchain in IoT networks. Ozyilmaz et al.[13] utilized a smart Gateway as one of the Blockchain node to integrate the Blockchain network with low-energy IoT devices. The Gateway facilitated a proof of concept and event-based messaging systems for resource constraint IoT devices to access Blockchain network. This research addressed the connectivity issue of IoT devices with Blockchain but high power, and bandwidth consumption required for the Blockchian remain unsolved in the proposal. Cha et al. [56] has developed a privacy preserving IoT framework,

which includes a Blockchain Connected Gateway (BC gateway) to incorporate the Blockchain network as the underlying infrastructure for privacy management. The BC gateway uses Blockchain technology to secure and track user privacy preferences. However, the research has been limited to address the user's privacy concerns.

Nonetheless, most of these proposals[13, 56] are at conceptual level and the notion of agent in continuously monitoring patient's health has not been still studied for the purpose of optimizing Blockchain algorithm and IoT eHealth data management. Health data is always regarded as a lucrative target for hackers and researchers are highly motivated to exploit the secure transmission and storage of protected health information (PHI). Recent proposals of building secure ehealth system adopts smart agent in the form of smart Gateway and smart contract to integrate Blockchain technologies in Body Area Sensor Networks (BASN). For example, Griggs et al.[35] integrated WBAN (Wireless Body Area Sensor Network) with a Blockchian network. Smart contract executed on the Blockchain can automatically analysis health data based on threshold values and record logs of transactions in an immutable ledger of the Blockchain for generating automatic reminders for care givers. However, in the existing research of IoT eHealth and Blockchain, little is known about the storage management of health data, mining management for the Blockchain and security and privacy of the patient end's devices. To bridge this research gap, we proposed a Patient Centric Agent assisted End to End decentralized Blockchain leveraged eHealth framework. The patient agent can provide high performance by integrating Blockchain, artificial intelligence and machine learning technology. The agent can address the challenges(as illustrated in Figure 1.7) raised while merging body area sensors with Blockchain.

1.3.1 To Overcome Resource Limitations

IoT devices are manufactured with limited computational power and memory capacities, while Blockchain technology requires an excessive level of storage and power. The resource requirements for mining Blocks on the P2P Blockchain network outweigh the capabilities of resource-constrained IoT devices. To cope with this challenge, the Patient Centric Agent running on Edge and Cloud server can handle Blockchain operations on behalf of the IoT devices. The Patient Agent runs a cost effective consensus mechanism and manages multiple Blockchains for IoT data.

1.3.2 To Reduce Bandwidth Consumption

Since the Blockchain maintains a decentralized ledger on a P2P network, the participants are required to broadcast Blocks throughout the network to include the Blocks in the distributed ledger and synchronizing it through executing a validation mechanism. IoT devices are equipped with limited bandwidth capabilities. Recently, Edge-devices augmented with IoT devices might provide sufficient bandwidth for the network traffic. However, the bandwidth required to operate Blockchain may exceed the upper thresholds of Edge servers. To deal with this, the Patient Centric Agent does not transmit transactions to the Blockchain network. Instead, it creates Blocks by organizing a certain number of transactions on patient providing devices. As a result, a significant number of transactions do not propagate throughout the P2P Blockchain network. Thus the inclusion of Patient Centric Agent with eHealth framework can reduce bandwidth requirements. Further, the Patient Centric Agent can optimize the consensus mechanism to reduce the bandwidth of the Blockchain network.

1.3.3 To Address Connectivity Challenges

In the P2P Blockchain network, all nodes remain connected to the network and autonomously operate using standard protocols. This nature of the Blockchain networks theoretically makes IoT devices more susceptible to security attacks. In our settings, the IoT devices are connected to Blockchain via the Patient Centric Agent which implements security protocols to safeguard IoT devices from cyberattacks. In addition, the Patient Centric Agent is replicated to a smartphone, Fog network and Cloud network to operate different subsets of BC operations on the three layers. As a result, IoT devices can connect to the BC leveraged Cloud storage via the secure Fog network.

1.3.4 To Address Memory Limitation

The Blockchain technology has had the most success in cryptocurrencies, where miners charge a fee for processing transactions without the aid of third parties. However, eHealth applications significantly differ from cryptocurrencies in the amount of storage required. A patient monitoring system continuously streams health data and transactions are frequently created. Storing all health data on-chain for many patients is challenged by the Blockchain's structure.

To meet this challenge, the Patient Centric Agent has been provided with the knowledge of determining rapid repositories for every data block based on data characteristics, patients' privacy preferences and features of multiple storage repositories. The data blocks which are low volume and require distributed processing platform are directed to distributed ledger of the Blockchain. Otherwise, other health repositories including Electronic Health Record, Electronic Medical record, Cloud eHealth are recommended as per the requirements of the data blocks. For example, billing documents, healthcare provider's notes, medication summaries that are less regularly generated in low quantities can be processed and stored in the Blockchain ledger whereas streaming health data such as normal ECG which requires huge volume of storage is transferred to Cloud based health repositories since there is virtually unlimited storage on Cloud servers.

1.4 Research Objectives and Questions

In this section, we describe two research objectives. Research questions are set to achieve the goals of the research.

1.4.1 RO-1: To Design a Patient Centric Cost Effective Secure IoT eHealth Framework

Internet of Things[57] applications such as remote patient monitoring (RPM) include wireless sensors that detect physiological signs with wearable sensors[58] and stream data to local and remote servers. However, wearable technologies introduce new security challenges[59] and their translation into health outputs in developing countries has been mixed[60]. Nevertheless, RPM[61] has been developed in a variety of applications such as continuous vital signs monitoring, arrhythmia detection, fall detection, regulating oxygen therapy, monitoring of pregnant women, chemotherapy reaction and glucose monitoring[62].

As data generated by RPM is health related, it needs to be stored at appropriate security levels while ensuring appropriate authentication, and access control mechanisms are in place. Ideally, RPM data should be integrated with other data including demographic, geographic location and

medical data and embedded within electronic health records. But government led electronic health record systems development tends to be enormously expensive and few countries have successfully implemented EHR systems despite the promise of efficiency and safety gains [63]. Progress towards the integration of digital health records into a consolidated virtual record of every interaction that an individual has with healthcare providers, has been checkered[64]. An electronic health record system introduced by the national government in Australia has cost well over $AUD 1 billion[65], a level that many national governments cannot not contemplate. Recently, Halamaka and Ekblaw[66] have advocated the use of Blockchain technology for the implementation of electronic health records that does not require enormous government investment levels.

However, as mentioned earlier, the integration of Blockchain into IoT network particularly in healthcare system raises privacy, scalability, throughput and storage challenges. The following research questions were designed to address the challenges of adopting Blockchain in eHealth and other IoT systems including underwater sensor and smart cities/home.

- Most EHR architectures are not designed for continuously RPM data streaming[67]. The archetypal RPM architecture involves sensors near, on or in the patient transmitting data wirelessly using Bluetooth, or ZigBee or customized protocols to a Base station forming a Body Area Wireless Sensor Network(BAWSN). The Base station processes data and transmits it to remote, often Cloud based servers for further processing[68]. Most RPM architectures use a central server for storage and processing of streamed data. However, a single server causes a single point of failure and bottleneck problems[69]. Further, patient's end devices depend on a trust center to implement user authentication and manage encryption keys. Reliance on a trust center introduces the possibility of delays in the transmission of patient's physiological data to a healthcare provider. According to Gemalto[70], the highest percentage of data breaches(around 34%) occur in the heathcare sector. Data stolen from a bank becomes useless once the breach is discovered and pass codes are changed. But data stolen from electronic health records might include personal identity and medical history that can impact a patient for years. This is a real problem as the leading cause of data breach in 2014 occurred from criminal attack[71]. Although EHR facilitates access to huge archives of patient history, privacy and data integrity protection is challenged when sharing a patient's record with many stakeholders [72]. If an EHR is compromised by attackers, the privacy of millions of users can be breached. These challenges lead us to formulate our first research questions:

  RQ-1: How to develop a robust and secured IoT RPM eHealth architecture to address patient's security and privacy?

  Fortunately, Blockchain technology has emerged as a secure storage technique that maintains a single version of the truth across a P2P network. The Blockchain applied in cryptocurrencies such as Bitcoin[73] and Ethereum [74] is a shared, tamperproof ledger and a peer to peer communication system where public/private key identifiers preserve user privacy. A universal set of tools for cryptographic assurance of data integrity, standardized auditing and formalized contracts for data access in the Blockchain ensures that healthcare providers and other authorized stakeholders can access patient records with permissions granted by the patient. EHR built on Blockchain technology could enable a comprehensive, interoperable, and secure exchange of patient records among different healthcare providers[75].

  Researchers[75–79] have sought to integrate Blockchain in eHealth system. Zhang et al.[76] presented a Pervasive Social Network(PSN) which included a security module placed at the

user end, which broadcasts transactions throughout the PSN to verify its signature using the master key of the sensor and the node itself. However, Blocks in Zhang's architecture are not used to store physiological health records but instead store patient or healthcare providers' meta data such as identity, address, and diseases.

The Health Care Data Gateway(HDG)[77] is a smart phone based App that integrates traditional database and a Blockchain distributed database to manage patient health data. HDG consists of three layers called the Storage Layer, Data Management Layer, and Data Usage Layer. Cloud is the platform for Storage Layer in Blockchain fashion. The mechanism that nodes use to validate a transaction known as the consensus mechanism in Blockchain algorithm has not been reported in HDG. Further, in HDG, a lot of Blocks and transactions are propagated in the Cloud P2P network which consumes higher bandwidth and power. Bowhead is a Blockchain based healthcare application[78]. The user can provide his information to Bowhead's application through different body area medical sensors and the application stores that information in a Blockchain based database. The Bowhead describes the procedure for collecting patients data, but it does not describe how the stream of medical data produced from a patient's body is embedded into the existing Blockchain.

MeDShare[75] is also a medical data sharing system among Cloud service provider via a Blockchain contract. The contract refers to a program written by a user defining terms and conditions of an agreement. In MeDShare, Blockchain nodes store the rules of the agreement. The MedShare authors discussed the system setup, requested file, package delivery, auditing and provenance in detail where the function of each layer of their architecture and smart contact technology in Blockchain are integrated to share data securely. However, MedShare constitutes only a sub-system in RPM architecture.

Zhao et al. [79] proposed a fuzzy vault based key management in a Blockchain health architecture. The architecture of the Blockchain based health framework includes wearable sensor nodes on the patient's body, some implanted nodes and gateway nodes for constituting BWSN. Zhao et al. [79] did not focus on patient's stream data management. Linn et al.[80] proposed to use an Off-Blockchain, a central database to store patient's health data called Data Lake, integrated with a Blockchain containing all authorization transactions for access control. Off data storage decreases the storage requirements that are required to be stored by each node on the Blockchain network. However, Off Blockchain might suffer from a single point of failure due to maintaining single storage medium. Further risk of off-chain data storage is that data accessibility is no longer assured as the data is not part of the Blockchain but only a footprint of the data is preserved on-chain[81].

The aforementioned eHealth architectures utilized the Blockchain technology introduced in digital currencies for health data management. However, most Blockchain architectures do not deal with continuous patient monitoring. The development of Blockchain eHealth architecture is challenging in remote patient monitoring due to huge amount of streaming data from wearable sensors, high computational costs and long transaction processing times required in Blockchain technologies. Health data can stream from sensors so rapidly that it can not be feasibly processed and added to a Blockchain in real time. Also, the volume of sensor data cannot be directly mined by Blockchain miners without risking falling behind schedule. Volunteer miners might be reluctant to join RPM Blockchain networks owing to the large storage and processing requirements. The existing state-of-the art research did not focus on meeting these challenges while applying Blockchain in IoT healthcare. The first research question aims to address the above mentioned challenges.

- Body area sensors upload their generated data to a central Cloud server through a Local Processing Unit(LPU) or Base station and patients share the health data in the Cloud with different stakeholders. Most of the conventional eHealth architectures are unable to meet the requests from an exponential growth of medical sensor devices predicted[82] and further raise scalability and interoperability limitations. Existing centralized body area sensor architectures are vulnerable to various malicious attacks including ransomware and Denial of Services(DoS)[83]. In addition, Cloud based storage and processing create concerns about patient's privacy as third party Cloud Service Providers(CSP) belong to these storage systems. Conventional CSP(Cloud Service Provider) cannot ensure accountability, and traceability of patient's medical data[84] as health data is stored in different off-premise Cloud servers.

  Further, Body Area Sensors have limited storage, processing and energy resources and cannot undertake the high computational power for processing Big health data. Mobile Cloud Computing(MCC) has emerged to expand the capabilities of Body Area Sensor Networks through data or task migration to Cloud servers. Task migration can overcome limitations of body area sensors such as limited memory, CPU power and battery life. Although Cloud servers support very large storage and very high processing capacity, excessive transmission delays and unstable connections can degrade the quality of service(QoS). If medical sensors directly connected to the Cloud become prevalent, transmitting and retrieving data to/from the Cloud can be expected to cause higher latency and become intractable. Recent advances in healthcare, Edge computing has enabled extensive processing capabilities at the Edge of the network. Edge computing can reduce this latency and improve quality of service because Edge devices are located closest to medical devices[85]. Medical data produced in settings such as emergency or intensive care units rely on rapid, near real-time transmission of data to healthcare professionals[86]. In these situations, the Fog resources closest to the Patient's Smartphone can support the processing of streaming data from wearable sensors in real time. Edge servers are now capable of extracting meaningful analytics from medical sensors to ensure a precise healthcare services. Recent eHealth architectures [87], [88],[85] incorporated Fog computing with smartphone and Cloud to make the processing of health data faster. Despite this ongoing advancement, there are growing concerns regarding the privacy and integrity of sensing and transmitting data to the Edge from its embedded medical sensors. These concerns prompted us to explore how to develop a Blockchain leveraged eHealth platform in the ecosystem of Edge and Cloud layers and the following research question has been set.

  RQ-2: How to develop a decentralized IoT eHealth architecture in multi-layers to withstand major cyberattacks?

  The rationale of adopting Blockchain into eHealth system is a trade-off between security and computational cost and storage resources. Body are sensors in eHealth have limited storage, processing and energy resources and cannot undertake the high computational costs, long delays and processing power required for the conventional Blockchain. Sensors cannot afford BC consensus mechanism which is the underlying core component of a Blockchain to ensure a common agreement about the Block's state and resiliency of the complete ledger. For instance, the Proof of Work (PoW) consensus mechanism used in [89, 90] requires high computational overhead, long delays, and a great deal of power. Researchers [90, 91] proposed several approaches to accommodate BC technology in body area sensor networks. Dwivedi [91] deployed a particular Gateway node to gather data Blocks from a group of

IoT devices and perform the verification function as a Miner before adding the Blocks to the Blockchain overlay network. This is far more efficient than a Proof of Work consensus but is vulnerable to numerous cyber-attacks. Tuli et al. [90] presented a generic Broker in the Fog to adopt Blockchain into Internet of Things data streams. Fog computing shifting the computational resources to the edge of the network can accommodate Blockchain operations which can increase response time, offer scalability, energy efficient and inexpensive deployments[92]. The broker assigns Blockchain tasks to various Fog devices so that the computational challenges can be met. However, this approach is still vulnerable to Denial of Service (DoS) attacks and tampering because they still rely on a centralized Blockchain controller[25]. Our research question is designed to deal with issues raised during adopting Blockchain in Edge and Cloud network.

- Huge amounts of health data are now generated which necessitates the diverse storage options. Worldwide, the total amount of digital healthcare data was 500 petabytes in 2012, and it is expected to increase in the amount to 25 Exabyte by 2020[93]. Further, new sensors(eg FitBit) enable continuous monitoring of physical signs and generate huge streams of additional data. Stranieri and Balasubramanian[94] estimated that a single patient produces around 300 megabytes per month data from vital signs sensors.

  Increasingly, patients have the option to choose to have their health data stored by diverse managers of storage media including: 1) public Cloud which is a powerful virtual information processing storage and can be accessed remotely from a Cloud service provider, 2) Blockchain storage which is a distributed, tamper proof shared ledger, not controlled by any entity 3) Healthcare provider managed Electronic Medical Record 4) Patient personal computer storage 5) Government providing Electronic health records, and others.

  Each storage repository has different security vulnerabilities and patients have diverse concerns about privacy. For example, hardware virtualization in Cloud eHealth facilitates sharing the same hardware among different users to execute various applications. However, Cloud eHealth is vulnerable to malicious attacks due to different unknown user's interaction including malicious attacks, trust management and non-repudiation among servers[95]. In addition, health data physically belongs to a Cloud provider once data is stored in the Cloud eHealth. This results in the risk of breaching patient's confidentiality and data integrity. Also, Cloud storage introduces delays in retrieving patient data. In contrast, Blockchain eHealth can withstand major cyberattacks and facilitate processing and storage of health data without the need of third party. However, Blockchain EHR also suffers from low throughput and limited storage capacities. In some settings (eg emergency) healthcare providers require rapid "Break the Glass" access to health-related data, ideally without compromising privacy. In this case, patient personal health repositories can preserve high security and confidentiality for patient's data but cannot afford high storage capacities. The dissemination of health data in multiple repositories can reduce the risk of patient's record keeping to a large extent.

  The type of health data is also now expanding. Health data stored in different repositories also varies from patient to patient in the level of sensitivity, and significance depending on medical, personal preference, and other factors. For instance, the continuous monitoring of patient's physiological signs has the potential to augment traditional medical practice in hospitals[94] and personal fitness[96]. The demand to store continuously streamed data has emerged recently, however this presents additional security, storage and retrieval challenges and further inhibits initiatives to integrate data to form electronic health record systems[89].

The problem of storing health related data is an important practical problem to tackle, as it has serious safety, privacy implications. Existing studies have addressed how best to select different Cloud storage services with respect to some performance and security criteria for user's data storage[97], [98] but the more general problem of which storage medium following data features including sensitivity, volume of data, patient's preferences, and QoS (Quality of Services) is best has not been addressed and solved. Besides this, little is known about the preferences an individual has for data stored by one agency over another. Knowledge of individual storage preference is particularly important for the storage of data streaming rapidly from wearable sensors where preferences need to be encoded into streaming software so that data can be channeled to the preferred medium in real time. Various characteristics of health data and the availability of a wide range of health repositories have inspired us to formulate the following research question.

RQ-3: How to determine an appropriate repository for health data based on data storage requirements while data is continuously streamed?

Patient's preferences regarding their privacy requirements can be expected to change from individual to individual based on the sensitivity of data and social context. For instance, a patient with a low public profile may be reluctant to desire especially high security for his or her ECG data whereas a patient with a high profile may require his or her ECG data to be stored extremely securely. Further, patient generally desires higher security and privacy for his or her psychiatric data than his or her blood pressure measurement data. Patient is also expected to desire more quality of services from healthcare professionals during high life-threatening situation rather than higher security and privacy. The diverse storage mediums also provide user with diverse level of CPU speed, capacity and disk I/O, networking latency and bandwidth.

Health data can be imagined to be disseminated among diverse agents managing storage repositories in such a way so that the nominated storage medium reflects data management requirements, including the quality of service, cost, volume, confidentiality, security and privacy of data that the patient desires for each chunk of data. So, the research question is how to develop a model that will take as input, user preferences under different contexts and suggest the storage medium with appropriate security, privacy and quality of service level as reflected in the user's preference.

1.4.2 RO-2: To Explore a BC for Managing Internet of Underwater Things and Smart home.

The Internet of Underwater Things (IoUT)[99] has enabled various agencies to track huge, unexplored underwater environments to reveal many precious resources including gas and gold, measure water temperature, observe fish , oil or gas pipelines and convey information pertaining to tsunami, water contamination or other natural disaster [100] that impact ecosystem of the earth.

IoUTs differ from their counterpart ground based IoT network in several ways[101]. IoUT devices typically deployed in hostile environment, and their unattended properties obstruct the development of secure and efficient management for the underwater ecosystem. Consequently, underwater communication is vulnerable to various malicious attacks[102]. Radio [103] is not suitable for underwater communication due to much attenuation. Acoustic signal (speed 1500ms^{-1}) used as communication channel features long propagation delay[101] and high bit error rates. Further, underwater communication channel is also attributed with low link quality due to the multipath

propagation and time variability of the medium. The above outlined attributes including large scale and sparse structure of underwater IoT present a significant challenge to design an effective secure routing protocol.

- Water current, low powered batteries which can be hardly recharged or replaced, limited memory and low bandwidth of IoUT devices construct extra barriers to develop a secured routing protocol to collect data from underwater IoT devices [104]. Various routing architectures and protocols including hierarchical (vertical) [105, 106] flat (horizontal) [107], location based [108], multipath routing [109], query based [110], and context aware [111] have been deployed in underwater IoT networks to transmit sensed data to a base station on the surface [112]. However, multi-hop based routing protocols that exchanged control message to discover forwarding path[113], [114],[115],[116],[117] developed for the transmission of underwater data to the surface drains more power of the node near the sink. Further, the key management and updating firmware of IoT devices for enforcing security in IoUT network consumes high energy which has not been fully addressed in the previous studies. Conventional IoUT routing protocols update security key and firmware with the aid of centralized authority and involve using many control packets that consumes higher power. The pre-condition of building a secure framework for such networks consisting of unattended devices at different layer is to devise a secure routing protocol. These issues motivate us to design the following research question under the second objectives:

  RQ-4: How to design a lightweight secure routing protocol for IoUT without the oversight of third party?

- IoUT architecture is comprised of interconnected underwater IoT devices that transmit data through perception, network and application layers[118] on the surface, and a remote often Cloud server. Most of IoUT architectures[111],[119], [120] rely on third party provided centralized server paradigm[121, 122] to analyze and store data generated by IoUT devices.

  The centralized IoT architecture is unable to efficiently handle large number of end to end communication from the massive numbers of underwater IoT devices. The entire system can be paralyzed by DoS or ransomware because the system represents a single point of failure and performance bottleneck that are are undesirable in a real time underwater system with a goal of high availability and reliability[8]. To address this issue, a distributed peer to peer wireless sensor network has been suggested to store and process IoUT data. But, security and privacy is challenged with a distributed peer to peer network while processing and sharing IoUT data. The research question below focuses on the development of an efficient secure decentralized framework for processing IoUT data.

  RQ-5: How IoUT framework can be developed to securely process underwater sensor data?

- An important application of IoT is smart home which refers to an integrated network that includes a range of home appliances such as TVs, light, fridges to provide real-time, smart services to customers without human intervention[123]. Users can exploit multiple home products to track and manage themselves depending on home network configuration. Many smart homes maintaining centralized networks can cause significant security vulnerabilities. In some cases, home appliances like smart TVs and refrigerators, which are the key components of smart homes, have been hacked to send malicious mails like phishing and spam messages. For instance, a child surveillance camera was hacked to make offensive

sounds in Texas, USA [124]. Smart home appliances are often exposed due to using unencrypted password and become the target of DDoS attacks. With the growing spread of IoT, the centralised IoT network structure poses the threat of numerous security vulnerabilities including data forgery, manipulation, and unauthorised access to devices by targeting Gateway services[125]. Therefore, smart home Gateway networks should be designed without the need for centralised system in an efficient and stable manner. Recently, Blockchain has been used in many next-generation applications and has become an appropriate method for providing security across a wide range of platforms, such as IoT, smart city, and many more[126].

IoT devices in smart homes require to communicate with various unreliable service providers over insecure channel. Preserving privacy of IoT devices in this application is paramount as well as saving the IoT framework from different cyberattacks is also important. The following research question targeted to explore a privacy preserving IoT framework for smart homes.

RQ-6: How privacy preserving IoT framework for smart home can be developed?

1.5 Contributions and Methodology

While Blockchain has emerged as disrupting services in Internet of Things, its implementation has a set of critical challenges in terms of scalability, high computational cost and privacy explained in Section 1.2.5. These issues raise while integrating Blockchain with Body Area Sensor networks. In this section, we describe our contributions and their methodologies.

1. Our first contribution to deal with the challenges of implementing a Blockchain for accommodating EHR of RPM system is to design a Patient Centric Agent(PCA) to connect Blockchain with the RPM data stream. The PCA is an artificial intelligent software agent that executes on a Patient's personal computer. The smartphone or Gateway devices playing the role of a smart Agent might be linked to multiple IoT devices and sensors. If the Gateway device is stolen or hacked, operations of multiple IoT devices associated to the Gateway might be affected due to various potential malicious attacks[127]. Furthermore, software Agent requires a platform that can support virtualization, data storage and high computational power for running accurate Blockchain algorithms and cryptographic algorithms. The Gateway device is unable to provide the Patient Centric Agent with appropriate execution environment. So, the software Agent is executed on a dedicated computer or Edge or Cloud servers which facilitate distributed environments. In general, the proposed PCA performs the following roles:

    - Ensure security and privacy at the patient's end.
    - Determine the storage and security requirement of streamed data. For instance, some streams will need to be stored in Blockchains, others can be archived with a lower level of security
    - Manage Blockchain providers. This includes selecting a Blockchain provider and facilitating insertion into a Blockchain by nominating a miner based on parameters including network latency, power consumption, availability, and trust.
    - Liaise with Trust Centers for key management

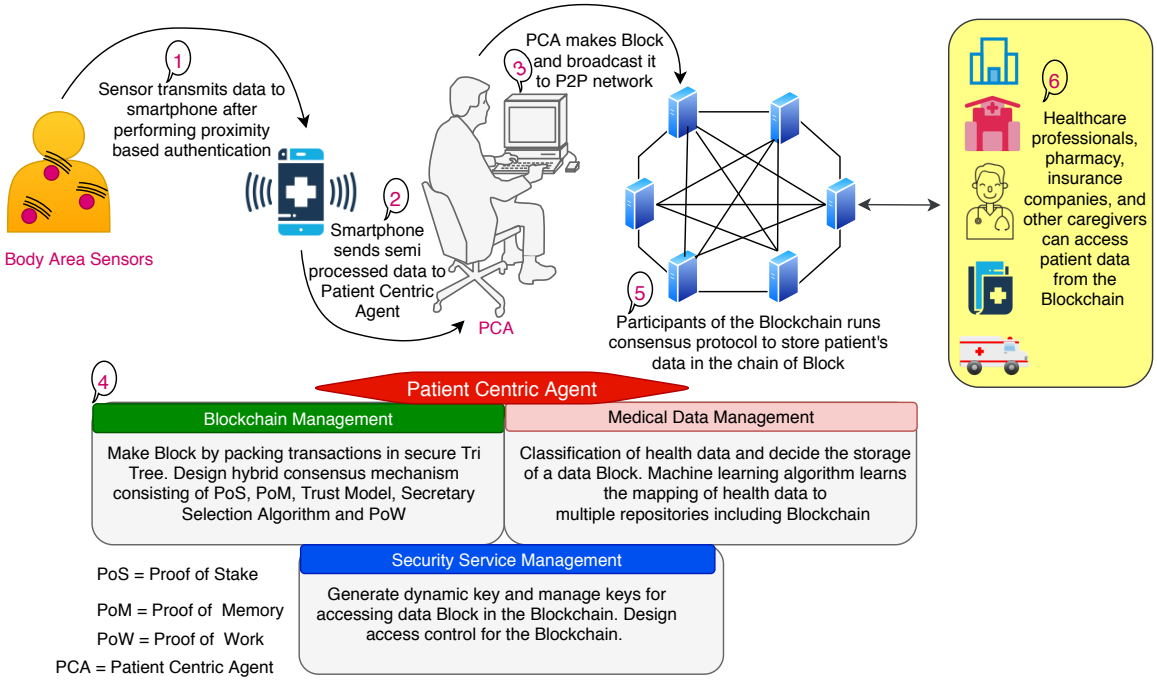


Figure 1.8: The basic operation of a Blockchain

In this dissertation, we first designed a continuous patient monitoring system that includes the proposed Patient Centric Agent for connecting the Blockchain with Body Area Sensor Networks. The PCA in the framework depicted in Figure 1.8 administers a portion of a customized Blockchain for implementing access control, running mining processing that includes selection process of Miners and managing multiple Blockchains to protect data privacy while streaming data from sensors. A lightweight communication protocol are introduced in the PCA-based architecture to improve data protection between different segments of the patient monitoring architecture in real time. Figure 1.8 suggests that the Patient Centric Agent running on a patient's personal device is placed in between smartphone and a customized private Blockchain to bridge two different networks(Body Area Sensor Networks and peer-to-peer networks). The bottom part of Figure 1.8 presents the main activities of the Patient Centric Agent: Blockchain Management, Data Management and Security Service Management.

We implemented a customized Blockhain using Java Programming to analysis the performance of the key algorithms designed in this proposal. The customized Blockchain was run several personal computers to analysis the performance using the NetBean. The high level analysis of the proposed eHealth architecture was performed in terms of end to end consumption, delay and major cyberattacks.

2. In the previous contribution, the Patient Centric Agent operates on a personal computer at the patient's end and collaborates between the Blockchain and the sensor networks. Consequently, the system has a centralised Blockchain controller at the patient's end, and a decentralised Blockchain storage at the other end. As a result, the patient's end is often vulnerable to major cyberattacks such as single point of failure (SPF) and denial of service (DoS). To solve this problem, we secondly contribute to decentralize the Patient Centric Agent by replicating the agent at Smartphone in Body Area Sensor Networks (Sensing Layer), Fog

devices (FAR processing Layer), and Cloud servers (FAR processing Layer). To process patient records rapidly, a lightweight modified Proof of Stake consensus protocol for the Blockchain is constructed using the Fuzzy Inference System (FIS). The consensus mechanism for data processing in a remote patient management is incorporated at the Fog layer. Further, the Patient Centric Agent replicated at the three levels enables outsourcing patient's tasks to Edge and Cloud nodes while preserving privacy and security. Decentralizing Patient Centric Agent in eHealth architecture results in software sustainability and allows the rapid and secure storage of medical data without the trusted authorities from third parties. The proposed decentralized eHealth architecture is presented in Figure 1.9. The Sensing Layer in Figure 1.9 includes various wearable sensors and smartphone to sense patient data. The next level of the sensing layer is NEAR processing layer shown as the middle part in Figure 1.9 consisting of Edge devices. The replicated Patient Agent in the Edge layer executes consensus mechanism for the Blockchain. The last layer consisting various Cloud service providers facilities high processing and storage for the Blockchain. The functionalities of this decentralized Patient Centric Agent shown in Figure 1.10 include four modules: Task Migration Handler, Blockchain Management, 5G Network Management and Security Service Management.

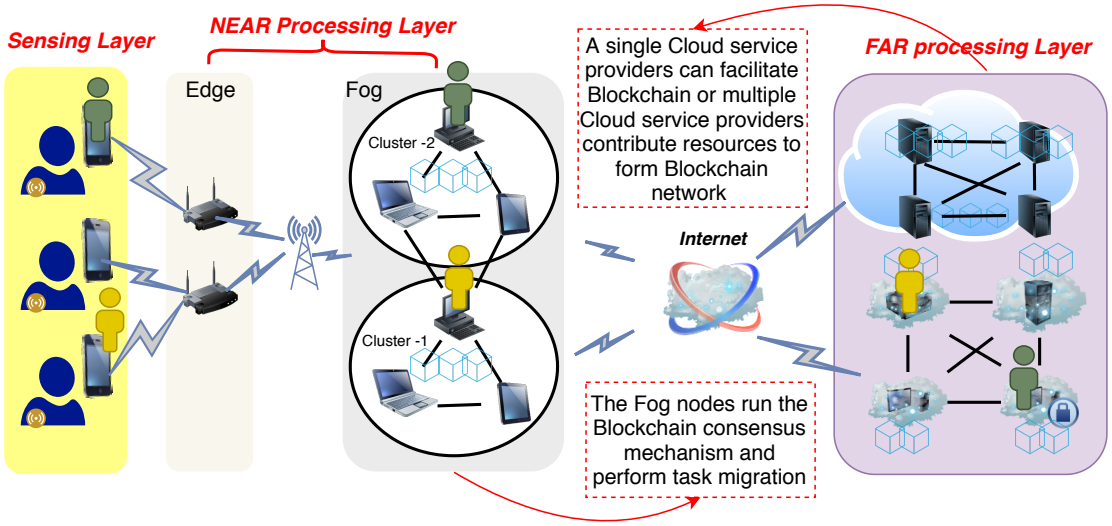


Figure 1.9: The decentralized Blockchain based eHealth System

We simulated the decentralized eHealth framework following the iFogSim[128]. The proposed consensus mechanism and privacy preserving task migration approach were implemented using Java Programming. The performances of key algorithms were analyzed in terms of Block generation time and energy consumption. The strength and reliability of the security protocols against major cyberattacks for the system was tested using Scyther [129]. To demonstrate the viability of the approach in eHealth monitoring, the comparison of the proposed frameworks with other existing systems was provided with respect to different metrics.

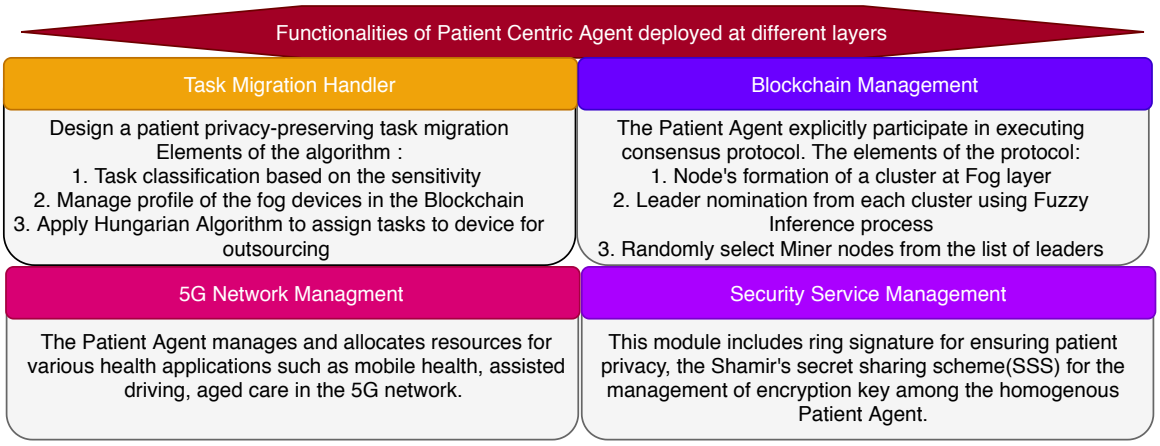


Figure 1.10: The functionalities of a decentralized Patient Centric Agent

3. A wide range of digital archives for storing health records has recently emerged. Some of widely used digital repositories include government-controlled Electronic Health Records (EHR), Electronic Medical Records (EMRs) maintained by health care providers, Personal Health Records (PHRs) operated directly by the patient, and modern Blockchain-based repository controlled mainly by technologies. These health record repositories differ from each other in terms of security, privacy, and quality of service (QoS) that they provide. The health data archived in these record systems often also vary in sensitivity, data importance, patient preference depending on medical, personal interest, and other factors including data volume, and data type. However, decisions about which digital record repository is most appropriate for the preservation of each data element at any point in time are complicated and nuanced. The health data continuously streamed from wearable devices escalate this challenge.

   To address this challenge, we thirdly contribute to enabling the Patient Centric Agent to build a machine learning based recommendation model for health data storage that can accommodate data storage requirements and patient preferences to make storage decisions rapidly, in real-time, even with streamed data. The rapid storage allocation model for health data is presented in Figure 1.11. The model depicted in Figure 1.11 has two parts: the upper part involves the processing of input and methods to constitute a training dataset, and the bottom part involves the machine learning approach. The diverse data blocks with different features and health repositories with their features are fed to the upper portion of the model as input. Several processes such as correlation coefficient analysis, heuristics rules, distance measurements and user preferences are applied to determine the repository for each data block.

   We generated a synthetic dataset having variable number of instances. The dataset represents data storage requirements and user's preferences regarding the archive of their health data. The four datasets have been fed into five different classifiers to study the feasibility of a machine learning algorithm in selecting an appropriate storage medium. Five different classifiers trained here are Multilayered Perceptions (MLP), Random Forest (RF), J48, K-nearest neighbor (IBK) and Naive Bayes (NB). The classifiers are trained using a variable size of the synthetic dataset in Weka ToolKits. The performance was analyzed with respect to accuracy, and root mean square errors.

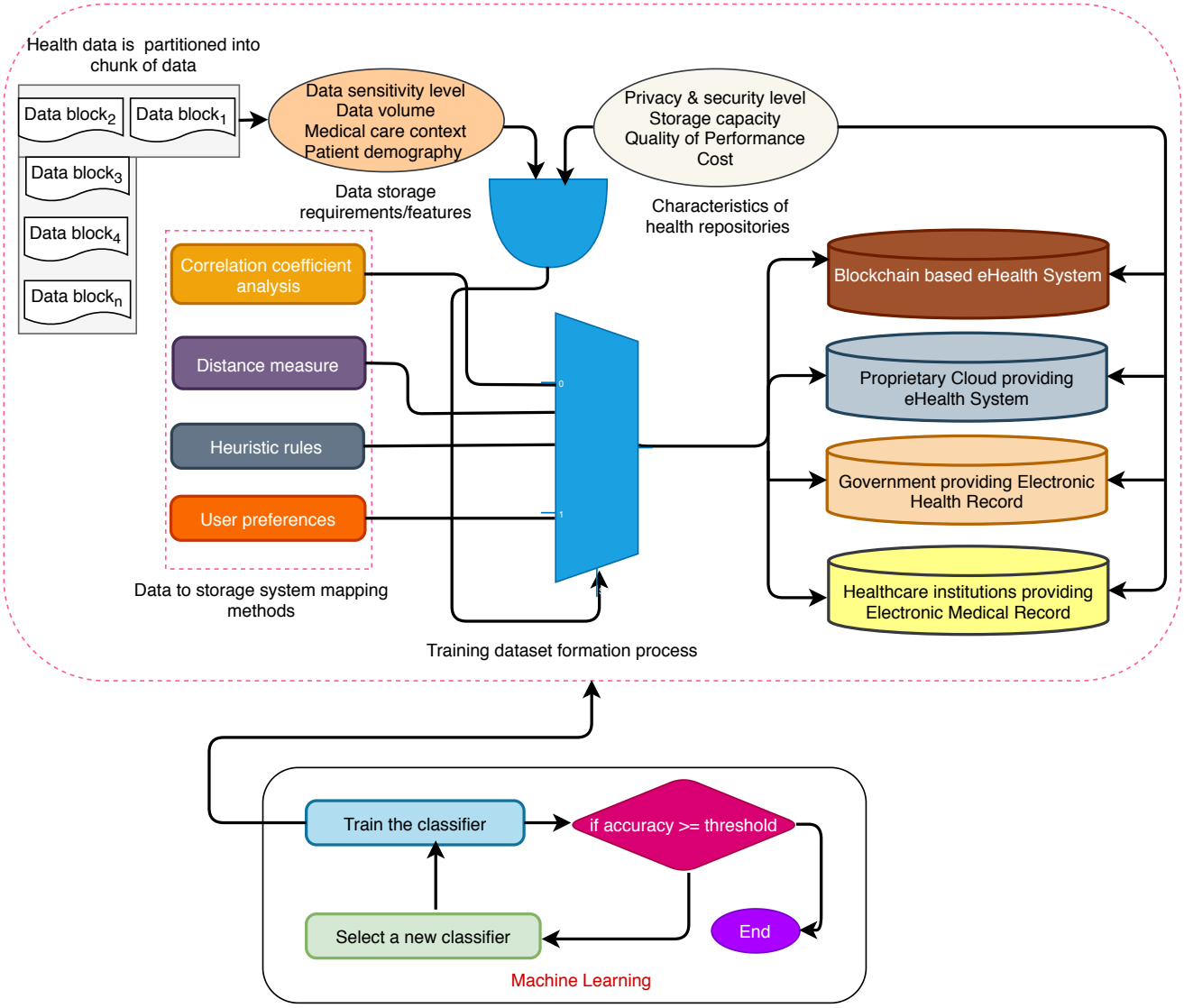


Figure 1.11: The machine learning based health data allocation systems

4. Our fourth contribution includes exploring the smart Agent's feasibility in tracking underwater IoT and IoT smart home or cities using a customized Blockchain. In Blockchain leveraged underwater IoT monitoring framework, we designed a secure light hierarchical routing protocol for the underwater sensors deployed at different depths and a lightweight consensus mechanism of the Blockchain for processing underwater IoT data. Underwater IoT maintains a lightweight Blockchain to manage their identities and security key whereas the Cloud servers form a peer to peer network to support another Blockchain which permanently stores IoT data.

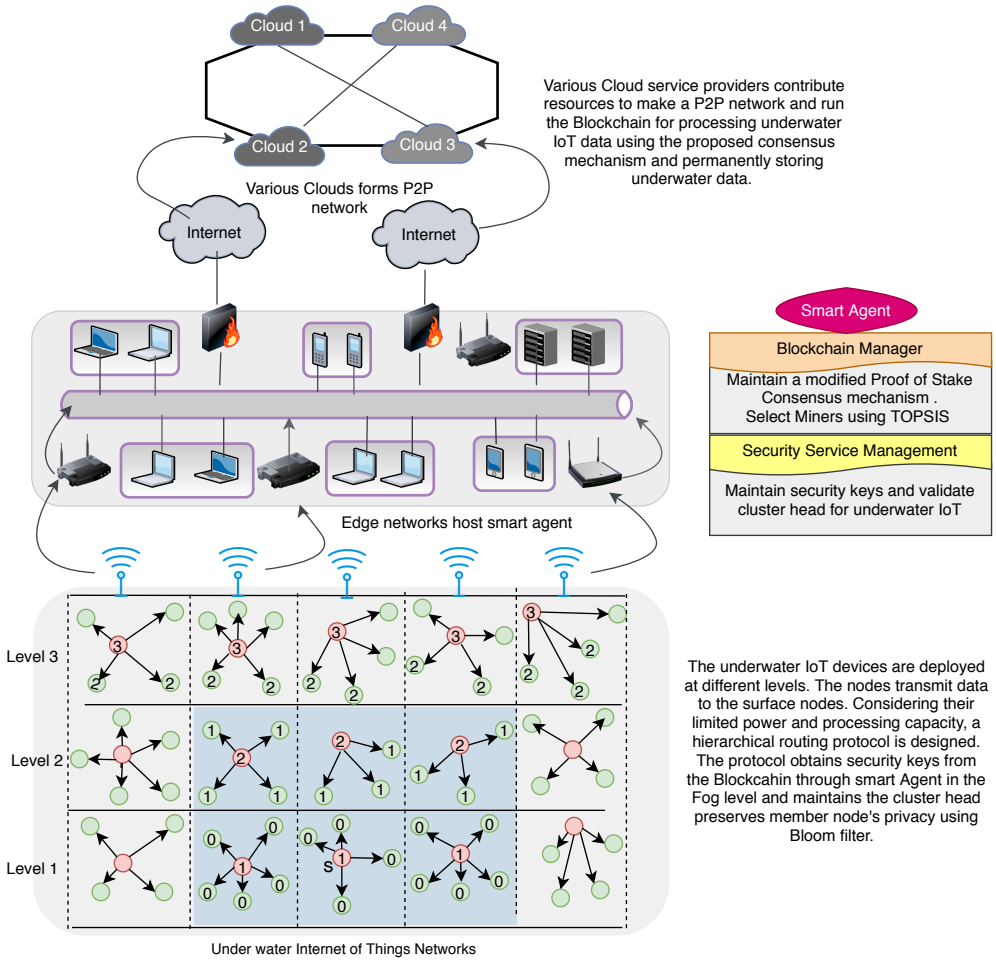


Figure 1.12: The Blockchain based underwater IoT monitoring framework

Java programming was used to implement the system. The lightweight framework for monitoring underwater IoT is depicted in Figure 1.12. The architecture depicted in Figure 1.12 consists of three layers: Underwater IoT layer, the Edge layer and Cloud layer. The smart Agent residing in the Edge layer receives data from the surface nodes of the IoUT layer and selects a group of suitable Miners from Cloud Blockchain network using TOPSIS method to process IoUT data. To analyse the efficiency of the proposed consensus protocol in detecting anomaly, we used publicly available datasets called KDD Cup 1999 Data[130]. In addition, the performance of the Blockchain-based routing protocol is evaluated in terms of different metrics such as block time generation, energy consumption, remaining energy and reliability.

5. Eventually, we have contributed to the creation of a IoT system to track smart homes or cities securely using Blockchain technology. In this architecture depicted in Figure 1.13, along with the smart Agent and the Blockchain component, we have included an extra Network Manager module to monitor IoT data packet using certificateless sign encryption that preserves user privacy. In certificateless sign encrypiton, IoT devices can transmit data to the smart Agent using their anonymous identifiers that are provided by the Network Manager. Similarly, the Blockchain Miner can play the role of Network Manager while the smart Agent sends data transactions to a destination node using the certificateless sign encryption method. As a result, the smart Agent's identity is not revealed to Blockchain nodes. As with previous contributions, we followed similar methodologies for implementing the framework. Few computers run the consensus protocol of a customized Blokchain. We use Jolinar [131] which is a Java software for estimating the power consumption of process level applications.

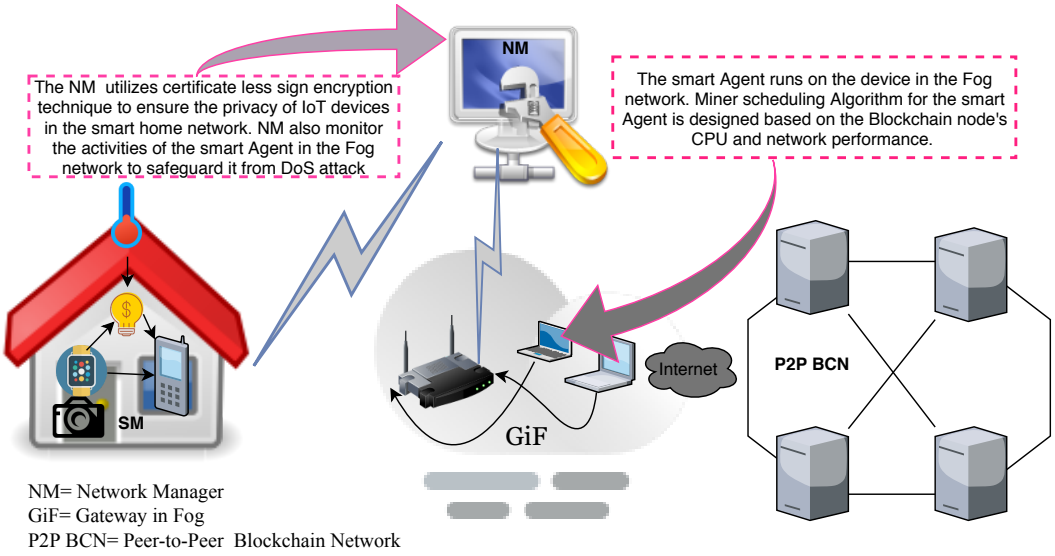


Figure 1.13: The Blockchain based smart home monitoring architecture

Finally, Blockchain can be optimized through two ways: the design of consensus mechanism and the underlying storage structure of the Blocks. The underlying Block storage structure entails that Blocks linked between them are stored in linear sequential orders(chain structured Blockchain) or graph structures(DAG(Distributed Acyclic Graph) structured Blockchain)[8, 132]. The throughput and energy consumption in the Blockchain can significantly improve based on the efficiency of the consensus protocol[16]. Therefore, in our proposed framework, we focus to develop efficient consensus protocols to increase throughput and reduce energy consumption for each contribution described above.

1.6 Organization of the Thesis

The remaining part of this thesis is organized as follows.

- **Chapter 2 Review of Previous Studies.** We review recent state-of-the-art studies. In this chapter, we present some recent Blockchain based IoT architectures that incorporated Fog, and Cloud computing. The previous studies are analyzed with respect to several factors. Further, we include the relevant literature to our research proposal in each chapter.

- **Chapter 3 A PCA Managed End to End Secure Customized Blockchain Based IoT eHealth Framework.** We describe the proposal of the continuous patient monitoring system with a Patient Cenric Agent, the implementation of functionality of the Patient Centric Agent and a customized Blockchain, and the evaluation through simulations using the Netbean Profile.

- **Chapter 4 The PCA Managed Customized Blockchain Leveraged Decentralized IoT eHealth Framework.** we describe the proposal of a decentralized Blockchain leveraged eHealth architecture, the extended roles(Blockchain based task migration, modified Proof of Stake) of the Patient Agent in this architecture, implementations of the framework using Java Programming and evaluation of the security protocols using simulators Scyther.

- **Chapter 5 The PCA Managed Rapid Storage Allocation of IoT Health Data with a Machine Learning Model.** we describe the proposal of rapid health data allocation using machine learning for the Patient Centric Agent, estimation model and the evaluation of the proposal using Weka Tools through synthetically generated dataset.

- **Chapter 6 The Smart Agent Managed Customized Blockchain Based Framework for Underwater IoT Monitoring.** We describe the lightweight Blockchain leveraged framework for underwater IoT, roles of the smart Agent to ensure security and privacy for the designed hierarchical routing, a lightweight consensus protocol, the use case of the consensus mechanism, implementation and evaluation of the framework.

- **Chapter 7 The smart Agent Managed Customized Blockchain Based IoT Framework for IoT Smart Homes.** We describe the design of a smart Agent based IoT home monitoring, the protocol for ensuring privacy of the users, the implementations and evaluations of the framework using Java Programming.

- **Chapter 8 Conclusions and Future Works.** Finally, in this chapter, we conclude this thesis with some limitations and future works.

Chapter 2

Review of Previous Studies

The emergence of paradigm Blockchain in Internet of Things (BCIoT) from the convergence of Blockchain (BC) and Internet of Things has spawned numerous smart services and applications bringing a lot of benefits in everyday life. Furthermore, the convergence of Blockchain technology with Fog and Cloud of Things have met the requirements of diverse applications. The combination has also resulted in other paradigms such as Blockchain in Fog of Things (BCFoT) and Blockchain in Cloud of Things (BCCoT) that can be incorporated into IoT infrastructure. Blockchain can address privacy and security issues in the Internet of Things (IoT) domain. The current IoT ecosystem has several security issues, including the deployment and management of Cloud servers by third parties, single point of failure, bottleneck and regular updates of firmware for millions of smart devices. The identified problems have inspired researchers to investigate Blockchain's adoption into the IoT ecosystem. In this chapter, we reviewed recent state-of-the arts BCIoT, BCCoT and BCFoT research in the context of eHealth, smart cities, and intelligent transport applications. In addition, we also identified the obstacles to embrace Blockchain in IoT and documented the research gaps and potential solutions.

The contents below of this chapter were published in the journal of Blockchain: Research and Application, Elsevier in February 2021. The article has already been cited 4 times (according to Google Scholar).
M. A. Uddin, A. Stranieri, I. Gondal, V. Balasubramanian, The adoption of Blockchain in IoT: Challenges and Solutions, Blockchain Research and Application,100006, ELSEVIER, 2021.

2.1 Blockchain, IoT, Fog, Cloud of Things, and SDN Paradigm

This section describes inherent issues of the Internet of Things, the Fog of Things, the Cloud of Things, and Software-Defined Network (SDN) with Blockchain's role as a panacea in these technologies. The studies reviewed in this article included Internet of Things, Fog and Cloud of Things with Blockchain technology to construct a framework for eHealth, wireless sensor network and smart home etc. The Internet of Things, Fog, Cloud of Things, Software Defined Network together with Blockchain technologies is described below before reviewing existing research from diverse domains that incorporated the technologies mentioned above. The list of acronym used throughout this chapter is presented in Table 2.1.

The chapter is organized as follows: the chapter starts with the basics of Blockchain technology. Section 2.1.1 provides an overview of Blockchain's fundamental components, and the

Table 2.1: The list of acronym

Acronym	Definition	Acronym	Definition
IoT	Internet of Things	BC	Blockchain
WSN	Wireless Sensor Network	M2M	Machine-to-Machine
CPS	Cyber-Physical Systems	QoS	Quality of Service
PoW	Proof of Work	BCIoT	Blockchain and Internet of Things
BCCoT	Blockchain and Cloud of Things	BCFoT	Blockchain and Fog of Things
PoS	Proof of Stake	BFT	Byzantine Fault Tolerance
PoA	Proof of Authority	PoET	Proof of Elapsed Time
SGX	Intel Software Guard Extensions	DDoS	Distributed Denial of Service
LPoS	Leased Proof of Stake	DPoS	Delegated Proof of Stake
DLT	Distributed Ledger Technology	P2P	Peer to Peer
EHR	Electronic Health Record	EMR	Electronic Medical Record
IoE	Internet of Everything	NOS	Network Operating System
CAT	Computed Tomography	TRL	Transaction and Read Latency
TRT	Transaction and Read Throughput	TL	Transaction Latency
RL	Read Latency	ABE	Attribute-Based Encryption
IPFS	Interplanetary File System	SAT	security access token
GDPR	General Data Protection Regulation	CSP	Cloud Service Providers
AHS	Artificial Healthcare System	API	Application Programming Interface
ARX	Add Rotate Xor	SVM	Support Vector Machine
VANET	Vehicular Distributed Ad-hoc	OBU	On Board Unit
CORE	Common Open Research Emulator	MAS	Multi-Agent System
RSU	Roadside Unit	AV	Autonomous Vehicle
NFV	Network Function Virtualization	SDN	Software Defined Network
CH	Cluster Head	SC	Smart Contract
IoUT	Internet of Underwater Things	PoBT	Proof of Block Trade
ACL	Access Control List	BASN	Body Area Sensor Networks
RFID	Radio-Frequency Identification	PCA	Patient Centric Agent
RPM	Remote Patient Monitoring	DAG	Directed Acyclic Graph
HLF	Hyperledger Fabric	LSTM	Long Short Term Memory
G2V	Grid to Vehicle	V2G	Vehicle to Grid
SWF	Simple Workflow Services	EVM	Ethereum Virtual Machine
VANET	Vehicular Adhoc Network	IoE	Internet of Energy

description of BC technology. Following this, the paper discusses the potential adaptability of Blockchain in Internet of Things, Fog, Cloud of Things and SDN technologies in Section 2.2, 2.1.2, 2.1.3, 2.1.4, and 2.1.4.1 respectively. The state-of-the-art works that explored BC and the Internet of Things, BC and Cloud of Things and BC and Fog of Things model in healthcare, supply chain, smart home, smart vehicular network, and miscellaneous IoT applications are presented in Section 2.2.1, 2.2.2, 2.2.3 and 2.2.4, respectively before concluding the article in Section 2.3.

2.1.1 Description of the Blockchain Technologies

Many research articles[133–135] partitioned Blockchain technologies into different layers. The section describes five layers of a Blockchain network along with the investigation of Blockchain's core properties related to immutability, security, and integrity. The layered structure of the BC depicted in Figure 2.1 is discussed below. We included basic background of Blockchain technology so that non-expert readers can comprehend this technology, and concepts and ideas of the thesis works. Readers who are familiar with BC technology can skip section 2.1.1.1.1, and 2.1.1.1.2.

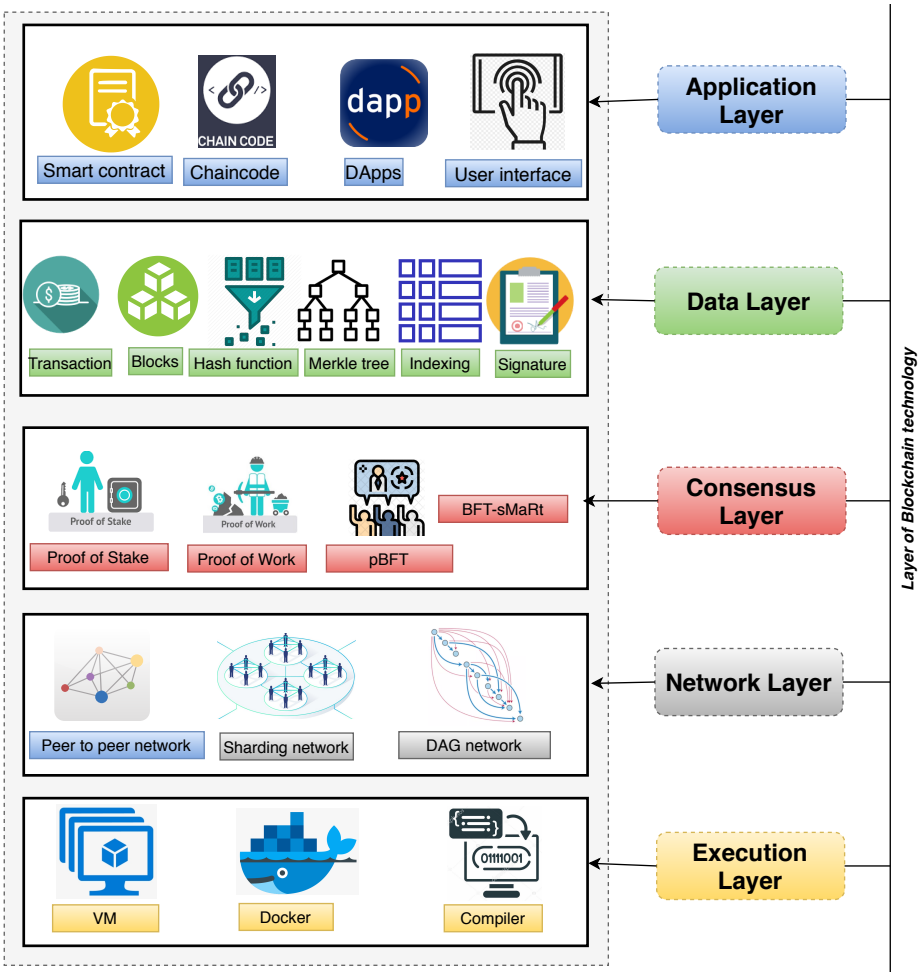


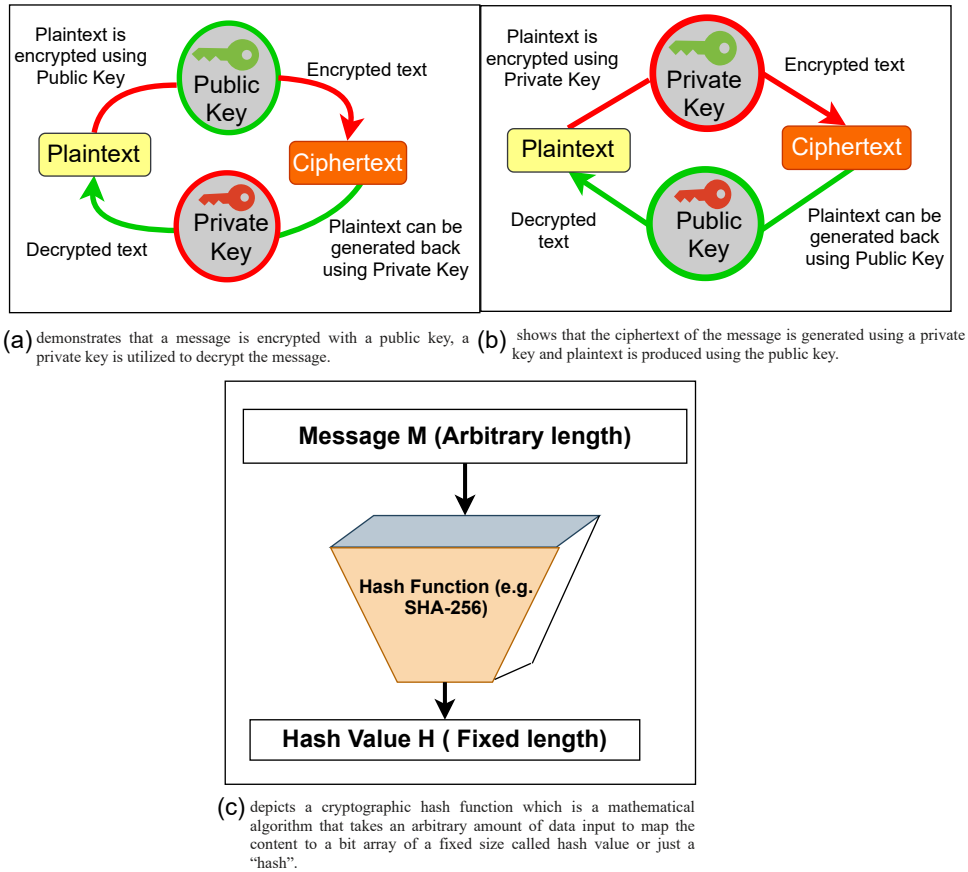
Figure 2.1: The layered structure of Blockchain technology

2.1.1.1 The Data Layer

This layer consists of transactions, Blocks, Hash function, Merkle tree, and digital signature. Transactions, Block and Merkle tree of this layer are discussed in the previous chapter. Digital signature is described below.

2.1.1.1.1 Digital Signature A digital signature[136] refers to a cryptographic approach to authenticate digital content and guarantee its integrity. Digital signature utilizes public key cryptography (PKI) system. Figure 2.2 shows the properties of PKI. Figure 2.2 (a) demonstrates that a message is encrypted with a public key, a private key is utilized to decrypt the message. Figure 2.2 (b) shows that the ciphertext of the message is generated using private key and plaintext is produced using the public key.

The public key of a user is known as his or her address like a bank account in BC technologies such as Bitcoin or Ethereum. Anyone can send digital currency to a user's address and only the user can access the currency using his private key of the corresponding public key pairs. Figure 2.3 explains signing a message using a user's private key and verifying the message with the user's public key.



(a) demonstrates that a message is encrypted with a public key, a private key is utilized to decrypt the message. (b) shows that the ciphertext of the message is generated using a private key and plaintext is produced using the public key.

(c) depicts a cryptographic hash function which is a mathematical algorithm that takes an arbitrary amount of data input to map the content to a bit array of a fixed size called hash value or just a "hash".

Figure 2.2: The properties of public and private key pairs

- Signing a message with a user's private key: To generate a digital signature of the message, the sender's signing algorithm produces a one-way hash of the message to be signed. A cryptographic hash function depicted in Figure 2.2 is a mathematical algorithm that takes

an arbitrary amount of data input to map the content to a bit array of a fixed size called hash value or just a "hash". The hash algorithm is a one-way function which is practically infeasible to invert[26]. The hash also known as digest is encrypted with the sender's private key. The digest along with other information such as the hashing algorithm is appended with the original message as a digital signature of the transmitted data.

- Verifying the message with the sender's public key: The receiver's signature algorithm verifies the electronic signature associated with the original content in two steps: 1) generating the hash or digest of the message 2) decrypting the appended digital signature using the sender's public key. If both digests are identical, the data has not been changed. Otherwise, either the message or signature has been altered or the digest has not been decrypted with the private key of the corresponding public key.

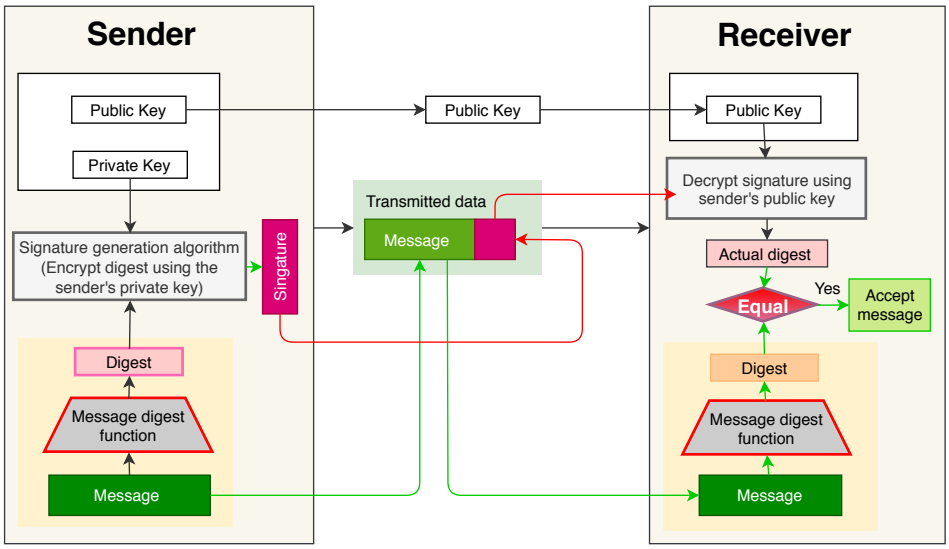


Figure 2.3: The processing of forming and verifying digital signature in Bitcoin BC

2.1.1.1.2 Different Types and Schemes of Digital signature In this section, we briefly discuss different forms of digital signature schemes with the merits and demerits of various technologies utilized to implement digital signature in Blockchain that are presented in Table 2.2.

1. Aggregate signature: The aggregate signature[137] is a traditional digital signature scheme based on co-GDH and bilinear mapping with an aggregation function. This scheme combines signatures of multiple documents into a single signature. For instance, users with public keys $PK_1, \ldots, PK_n$ sign messages $M_1, \ldots, M_n$ and creates signatures $s_1, \ldots, s_n$. Using aggregate signature scheme, signatures $(s_1, \ldots, s_n)$ can be compacted into a tiny signature s. This single signature can be verified using respective set of public keys to check integrity of messages$(M_1, \ldots, M_n)$. Aggregate signature can address the issue of limited storage and bandwidth.

2. Group signature: A group signature scheme[138] is a method of enabling a member of the group to sign anonymously on behalf of the group and in special cases, provide the

possibility of tracing the identity of the signer. A participant in the group can verify the signature using the verification key that it was indeed created by someone in the group but cannot discover who creates the signature. The authority can track the signer back in the event of conflicts or misbehaviour using the tracing key. Helix Blockchain implemented group signature for ensuring that transactions are ordered in a fair way in a Block.

3. Ring signature: Ring signature schemes[139] enable the participants to sign a document in an anonymous way on behalf of a spontaneous group. The ring signature scheme, unlike group signature, does not need the group manager to construct the group or allocate keys to members of the group. The signer, in other terms, will spontaneously create the group without the assistance of other group members. Several cryptocurrencies including Bytecoin, ShadowCash, Monero, Verge, Zcoin, and Dash implemented ring signature to preserve users' privacy.

4. Blind signature: Blind signature[140] is a form of digital signature that blinds the document before signing it. The signer will therefore not know the content of the document. A variety of public-key encryption schemes can be applied to create blind signatures. PayCash and Moneta Express have already implemented blind signature in their payment system.

5. Proxy signature: A proxy signature scheme[141] enables an entity known as the designator or original signer to delegate to another entity called as a proxy signer to sign messages on its behalf in case the original signer is unable to sign due to temporal absence, lack of time or processing power.

Different digital signature algorithms vary in the technique of generating public/private keys. Several schemes of forming digital signature is presented in Table 2.2 and 2.3.

Table 2.2: Different schemes to from digital signature

Digital Signature	Description	Merits and demerits
RSA[142, 143]	This signature scheme is based on the RSA cryptography. The strength of RSA is derived from the computational complexity of factoring large integers which are the multiplication of two large prime numbers.	• Key distribution is convenient • Smaller numbers of key are required for large network compared to symmetric key • Low operating speed and high computational cost • Vulnerable to multiplicative attacks
ECDSA[143, 144]	Elliptic Curve Cryptography (ECC) is an alternative to RSA for digital signature development based on elliptic curve theory that produces quicker, smaller, and more powerful cryptographic keys. The algorithm's strength levels derive from the problem of solving the discrete logarithm in the elliptic curve point group.	• Faster, smaller, and powerful • No application-based performance issues • A little chance of identical signature for two different contents
ElGamal Encryption System[145]	The security of this technique stem from the complexity of computing discrete finite field logarithms. The ElGamal encryption system encompasses both encryption and digital signature algorithms.	• Providing high level of security because of probabilistic nature • Facilitating digital signature for large numbers using a single key • longer computing cost for doubling the length of texts
DSA[146]	DSA is a Federal Information Processing Standard for digital signatures, based on the mathematical concept of modular exponentiation and the discrete logarithm problem. DSA is a variant of the Schnorr and ElGamal signature schemes.	• Lower computational costs and storage space • Complicated remainder operators for verifying signature

Table 2.3: Different schemes to from digital signature

Digital Signature	Description	Merits and demerits
GOST R 34.10-2012[147]	This is the Russian standard algorithms for generating and verifying digital signature based on elliptic curves.	Recommendations for curve uses are not required provided that only a set of requirements for such curves is needed
Schnorr Signature Algorithm[148]	This is a variation of the ElGamal encryption system and the FiatShamir scheme	Smaller signature size
Rapid Digital Signature[149]	This underpins BLS, DiffieHellman, and the Fiat-Shamir scheme.	• Simplified computing, pushing up performance levels • limited to groups with the pair matching function
Rabin Crypto system[150]	Security strength stems from the difficulty of integer factorization	• Higher operating speed • Susceptible to an attack based on the selected ciphertext

2.1.1.2 The Consensus Layer

No centralised organization is empowered to monitor the transaction or prevent attackers from changing or altering data when a node exchanges data on the Blockchain network. To combat fraud-related activities such as double-spending attempts, the Block's integrity must be verified, and the data flow must be managed to ensure a smooth exchange of data[27]. Validation mechanisms known as consensus algorithms are used to meet these objectives. A consensus algorithm is a means of obtaining an agreement between several insecure nodes on a particular data block in the Blockchain context. Several consensus mechanisms from the literature are described below and presented in Figure 2.4 which shows five categorizations of consensus mechanism: Proof of Work (PoW), Proof of Stake (PoS), Byzantine Fault Tolerance (BFT), Proof of Authority (PoA) and Proof of Elapsed Time (PoET). Protocols pertaining to our contributions are described in the previous chapter. In this chapter, we presented the rest of the protocols shown in Figure 2.4 below.

1. **Proof of Bandwidth:** In this process, the miner is selected and rewarded based on the bandwidth they contribute to the network. However, malicious nodes can falsely report their bandwidths. Therefore, a bandwidth measurement scheme is adopted to estimate the bandwidth each participant contributes to the Blockchain. Blockchain nodes can evaluate and measure each other's bandwidth contribution to reach an accurate consensus about relaying bandwidth. This approach can resist attacks that occurred by malicious nodes' colluding[151].

2. **Proof of Authority:** Proof of Authority(PoA)[152] is a consensus protocol that provides a small and designated group of Blockchain actors with the power to validate transactions. The PoA protocol leverages authorities' trust, which indicates that Block validators are not

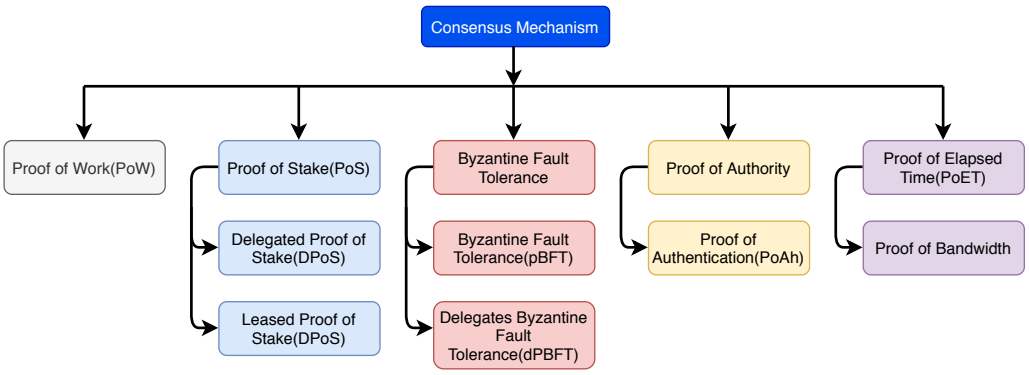


Figure 2.4: The taxonomy of consensus mechanism

required to stake coins; instead, they stake their reputation to the system. The PoA is applicable for private Blockchain and scalable since the limited numbers of pre-approved validators.

3. **Proof of Authentication[153]:** In Proof of Work, the first step a miner performs is to validate the Block, followed by calculating the target hash value of the Block. Conversely, Proof of Authentication[154] intends to authenticate the Blocks by verifying the Blocks' transactions according to PoW. In Proof of Authentication, a small group of trusted nodes are selected to confirm the Block and then add it to the distributed ledger. The authentication process involves two steps: verifying the source of the Block and increasing the point of each node that performs the authentication by one as its reputation. Every time a node conducting false authentication loses a specific unit of trust value and is reported as a regular node after a certain number of invalid authentications have been performed by it. Finally, the validators broadcast the Block throughout the network for all the nodes to update the distributed ledger. Proof of Authentication is deemed as appropriate consensus protocol in IoT as it avoids the inverse hash computation for energy-efficient distributed secure communications and computing in IoT[154].

2.1.1.3 The Network Layer

The network layer, also known as the P2P network, establishes communication between nodes. The P2P network ensures that all nodes can discover and connect each other to propagate Blocks throughout the network and synchronize the valid, current state of the Blockchain. A P2P network is a network of computers where computers (nodes) are distributed, and the workload of the network is shared across multiple nodes to achieve the end target nodes on the Blockchain for processing transactions and Blocks. Two kinds of nodes are maintained in the BC peer to peer network: the full node and the light node. Full nodes ensure that transactions and Blocks are checked and validated using rules prescribed in the consensus mechanism, which is also called mining. Full nodes are accountable for holding trust in the network, whereas light nodes can make transactions and send those to the full node. Light nodes can only store the header of the Blockchain (keys) while the full nodes store the complete distributed ledger.

Sharding: Sharding[155] that partitions a peer to peer network is introduced to improve Blockchain's performance. Sharding is a splitting strategy that distributes computing and storage workloads across a P2P network such that unlike conventional BC, each node is not responsible for managing the entire network's transactions load, but instead handles information related to

its partition or shard. Figure 2.5 presents an example of BC sharding. In this technique, several Blockchains called a chain of a shard are managed by network nodes instead of maintaining a single Blockchain for all transactions. Each shard consists of its own nodes or validators that apply a PoW or staking or voting consensus mechanism. Readers are suggested to go through [155–157] to have comprehensive knowledge on BC sharding.

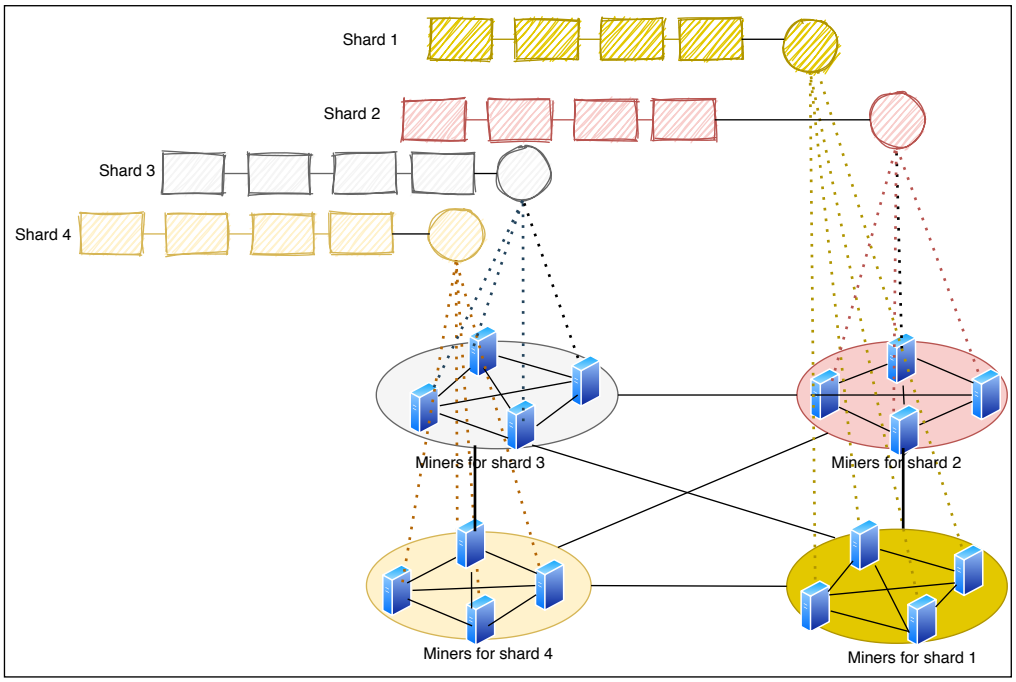


Figure 2.5: Example of a BC sharding

2.1.1.4 The Infrastructure Layer

We describe the infrastructure layer of Blockchain technology with respect to two enterprise BCs: Ethereum and Hyperledger Fabric.

A user's computer can participate in Ethereum Blockchain by running a client software such as Geth, Parity or Pantheon. Ethereum maintains two kinds of nodes: light node, and full node. The light node runs the client software stores the cache, stores the state of the Ethereum. The light node engages in verifying the execution of transactions while the full nodes download the entire ledger to their local storage, participate in full consensus enforcement, verify signature, transactions and Block formats and double-spending. The Ethereum nodes execute the Ethereum Virtual Machine (EVM) which is like Java Virtual Machines (JVMs) can run byte code. EVM acting as sandboxes offers an execution environment for a smart contract. EVM is a Turing complete software; a stake-based virtual machine that handles the internal state and computation for smart contract.

The Hyperledger Fabric BC is comprised of three types of nodes: endorsers, orderers, and peer nodes. The peer nodes host ledgers and chaincode (also known as smart contracts). The uses' applications and administrators using Fabric Software Development Kit (SDK) APIs can always communicate with peer nodes to access the chaincode or distributed ledger. The Hyperledger Fabric manages multiple channels that refer to sub-network (Private) can consist of a number of peers(member). Each channel maintains its separate ledger which is stored in each peer on the

channel. A specific set of applications and peers can communicate via channels. The transactions flow in the Hyperledger Fabric in the following three phases.

- Endorsing phase: First, the endorsing peers receive update transactions from an application. These nodes endorse update transactions without committing it in the ledger. They send the endorsement of the transaction to the orderer nodes.
- Ordering phase: The orderer nodes collect endorsed transactions from the endorsing nodes for various applications. These nodes order the transactions into Block.
- Distribution phase: Finally, the Block is distributed to all the peer nodes on the BC business network. These peers will validate the transaction and will commit the transaction to their local copy of the ledger upon successful validation.

The components of this layer are listed below.

- **Smart Contract:** Smart contract[158] written in Solidity language runs on the Ethereum runtime engine. The compiler produces bytecode of a smart contract that runs faster on the EVM. Code executed on an EVM is isolated from the network or file system. A smart contract refers to a set of business logic presented in various functions that are executed when a transaction against those functions is issued. The bytecode of a smart contract is assigned a unique address after deploying it on the EVM. A transaction associated with a smart contract can result in a state change in the decentralized ledger. Figure 2.6 is an example of a smart contract application for managing the trust for e-commerce sites. Many studies reviewed in this paper used a smart contract for different purposes in IoT applications as provided in Table 2.4.

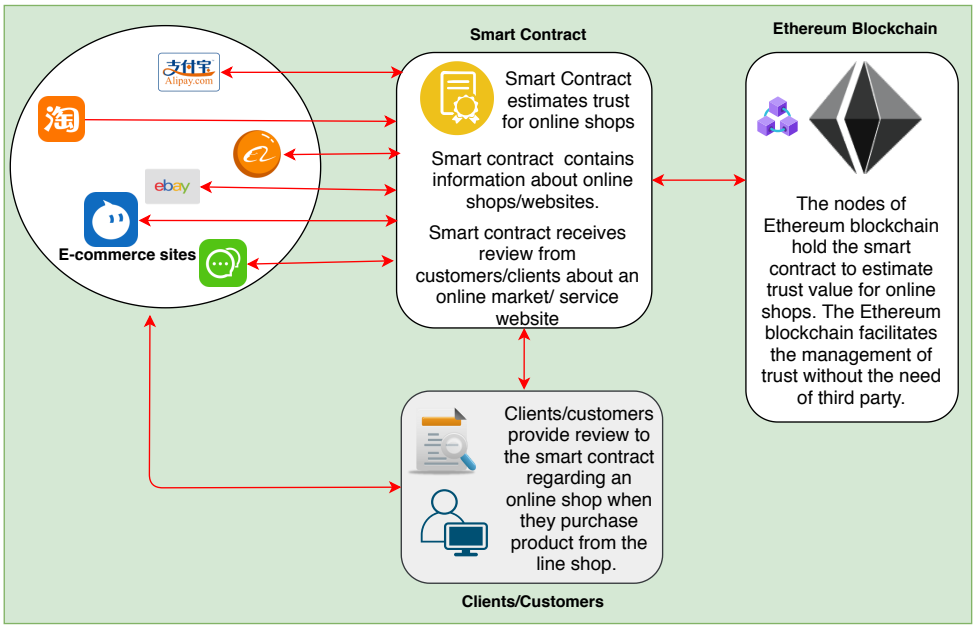


Figure 2.6: Example of a smart contract application

- **Chaincode:** In Hyperledger Fabric, several related smart contracts are packaged together into chaincode that is deployed in the BC business network. For example, an insurance

Table 2.4: Smart contracts in different IoT applications

References	Purpose	Applications
[159] [160][161][160][162]	Access control	eHealth
[75][161] [36]	Tracking access behaviour, access policies	eHealth data sharing, Edge network
[163][164] [165] [166][91][167]	Store sensor data	Body Area Sensor Networks
[168, 169]	Crowdsourcing	eHealth
[170] [171][172] [173] [174]	Incentive and payment management	EMRs, IoT smart cities
[175]	Enrolling patients and healthcare professionals	Remote patient monitoring system
[176] [177][178][179]	Authorization	Medical Forensics, Edge services
[180] [181] [35]	Maintain log information, auditing, analyzing	Biomedical queries, IoT
[182][183]	Maintaining policies for updating firmware	Vehicular network, supply chain in IoT
[184] [185]	Managing node's reputation	IoT ecosystem
[186]	Resource management in Edge network	SDN-IoT ecosystem
[187][188]	Detection of malicious activities	SDN-IoT ecosystem
[189]	Energy management	Smart grid
[190]	Trust management	Edge-Cloud network

application requires to implement their business logic in the form of multiple smart contracts named as claims, liability, processing, and so on, which together constitute a chaincode. The chaincode governs packaging and deployment of smart contracts in the Hyperledger Fabric. Further, chaincode defines the schema of ledger's data, initiates it, performs updates to ledgers based on consensus, and responds to queries for ledger data.

Unlike EVM, in Hyperledger Fabric, chaincode written in standard languages such as Java, Node.js and Go is deployed on peer nodes owned by different organizations. The chaincode runs on a secure Docker container. The client applications can access to chaincode via REST APIs or SDK. Chaincodes are initiated for a particular channel where an administrator determines endorsement policy for a chaincode running on the channel.

2.1.1.5 The Application Layer

This layer comprises two sub-layers: 1) presentation layer and 2) execution layer. The presentation layer includes scripts, APIs, and user interface. These tools are used to connect the application layer with the Blockchain network. The execution layer includes smart contracts, chaincode and underlying rules. The presentation layer sends instructions to the execution layer, which runs transactions. For example, instructions are sent to chaincode in Hyperledger Fabric and the smart contract in Ethereum Virtual Machine.

dApps: dApp refers to a distributed web application that runs on top of a distributed Blockchain technologies such as Ethereum, Bitcoin, and Hyperledger Fabric. dApp can interact with Blockchain using smart contract or chaincode. Unlike a conventional app, dApp is no longer controlled by a single entity or an organization once it is deployed on the BC network.

2.1.1.6 Types of Blockchain Technology

Figure 2.7 shows a classification of decentralized ledger technology (DTL). DTL in the literature differs with respect to data structure and accessibility.

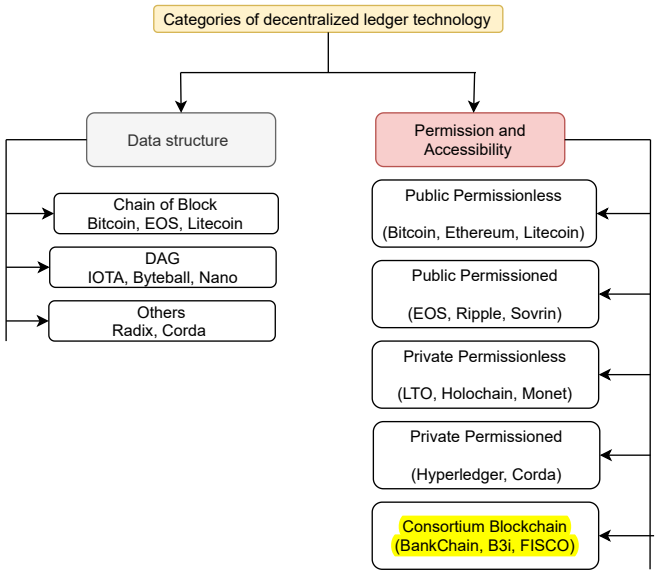


Figure 2.7: The types of decentralized ledger technology

In chain structured DTL, Blocks are linked between them in linear sequential orders while graph-structured DTL stores transaction in a Distributed Acyclic Graph[8, 132]. Individual DAG transactions are directly connected to each other rather than joined together and processed in Blocks. Depending on the accessibility, BC can be further categorized into two major types: public Blockchain (or permission-less) and private Blockchain (or permitted). A public Blockchain is a non-restrictive, permission-less distributed ledger system that allows anyone to join the network and make transactions as well as engage in the consensus process[35]. Bitcoin and Ethereum with open source nature and smart contracts are the most prominent public Blockchains. Public Blockchains are mostly reliable if the users strictly abide by the rules and regulations of the Blockchain[36]. On the other hand, private Blockchain is an invitation-only network operated by a central authority, and a validation process would allow participants to confirm transactions in the Blockchain.

However, a group of the Blockchain developers debate that private Blockchains cannot be considered as Blockchain as the principle of monitoring, tracking, and restricting the number of participants in the private Blockchain contradicts the trustless and open nature of the Blockchain[191]. Private Blockchain differs from public Blockchain in many aspects. The validators in public Blockchain are unlimited and can not be trustworthy whereas a premeditated number of validators process transactions in the private Blockchain which results in higher throughput and ensure strong privacy of users' data on the private distributed ledger. If a transaction is submitted on a public Blockchain, the transaction is tamper-proof and can never be altered or modified while a committed transaction can be updated in a private Blockchain following consensus of a certain number of authorized participants.

To set up a network, public Blockchains require no infrastructure costs, while private Blockchains need wide-scale deployment and operational costs[192]. Rimba et al.[193] compared the computation and storage cost of a Blockchain process with traditional Cloud system. They run two instances of business process from two different kinds of infrastructure: Ethereum Blockchain and Amazon Simple Workflow Services (SWF) to estimate costs of their business process logic. Rimba et al. reported that the cost of execution of the business process on Ethereum Blockchain could be two orders of magnitude greater than on Amazon SWF (Simple Workflow Service).

Another important Blockchain type is called consortium Blockchain, which is a semi-decentralized and governed by a group rather than a single entity. Variations of these kinds of Blockchain applied in the existing research articles are presented in Table 2.5 and 2.6 respectively.

Table 2.5: The different types of BC in IoT literature

Acronym	Explication	Interpretation
PuB	Public Blockchain	Each of the transaction in a public Blockchain is open for the public to verify. Anyone can download BC protocols and read, write or participate in the network.
PrB	Private Blockchain	The private Blockchain allows only trusted parties to participate in the network to verify and validate transactions.
CoB	Consortium Blockchain	The consortium Blockchain is a semi-private which is controlled by a group of users across different organizations.
EEB	Enterprise Ethereum Blockchain	Ethereum is the second-largest enterprise open-source Blockchain which is used for general purposes. Ethereum facilitates smart contracts and Distributed Applications (DApps) to be built and run without the requirements of a third party, any fraud and downtime.
PrEB	Private Ethereum Blockchain	Ethereum Blockchain network describes a set of nodes connected to each other to create a network. Developers can build a private Ethereum network rather than the public network to make transactions and build smart contracts without the need for real Ether.
EHF	Enterprise Hyperledger Fabric	Hyperledger Fabric refers to an open-source, permissioned distributed ledger developed by the Linux Foundation-hosted Hyperledger consortium. The client application uses Hyperledger Fabric SDK or REST web service to interact with the Hyperledger Fabric network.
PuPB	Public Permissioned Blockchain	A Public-Permissioned Blockchain network is defined as a new kind of network that bridges the gap between the Public-Permissionless networks (such as Bitcoin or Ethereum) and the Private Consortium networks.
PrPB	Private Permissioned Blockchain	This Blockchain is permissioned and private, so only selected participants can join the network. (e.g., Hyperledger Fabric, R3's Corda).

Table 2.6: The different types of BC in IoT literature

Acronym	Explication	Interpretation
CuB/ CPuB/ CPrB	Customized Blockchain/-Customized Public Blockchain/Customized Private Blockchain	Developers or researchers use popular programming languages like C++, Java, Python, Go language to build their own private or public Blockchain for analyzing the performance of their applications.
EPB	Enterprise Permission Blockchain	This is industry level Blockchain such as Hyperledger Fabric where users require permission to participate in the network.
CB	Cloud Blockchain	Third-party Cloud such as AWS provides resources for building and operating Blockchain operations.

Sidechain: The sidechain [194] refers to a separate Blockchain which operates in parallel to the main Blockchain and attached to the main chain by means of a two-way peg. The parent chain is called the original or main chain, and all additional chains are referred to as side chains. The two-way peg depicted in Figure 2.8 is a bidirectional transfer mechanism which enables users to move digital assets to the side chain from the main Blockchain and vice-versa. A user on the main chain requires to send a certain amount of digital coin to an outside address of a system called Federations. The equivalent coin is released on the sidechain after waiting for a certain time of the transaction committed. The user can access and spend the digital coin on the sidechain. The reverse occurs when switching back from a sidechain to the primary chain. A federation is an intermediary point for determining when to lock and unlock digital coins between the main chain and side chains. The federation adds an extra layer between the main chain and the sidechain. The developers of the sidechain might choose members of the federation. A sidechain with its own protocols and implementation can independently run and is completely isolated from the main chain. As a result, if the main chain is hacked or compromised, the sidechain can still operate likewise, if the cyberattacks on the sidechain cannot affect the operation of the main chain.

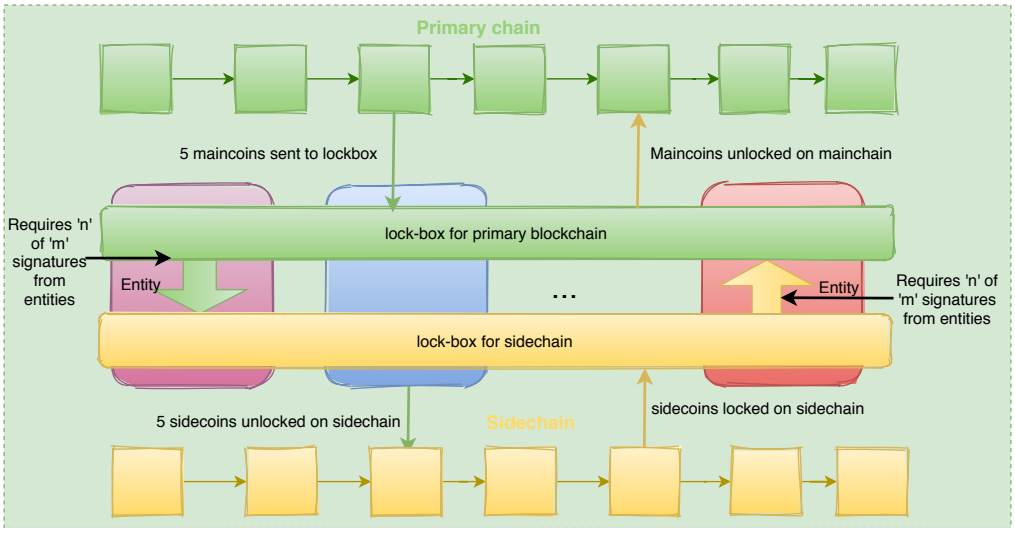


Figure 2.8: The Federated two-way peg communication

The sequence diagram of communications between the Main chain, Federation and Sidechain

is presented in Figure 2.9 where:

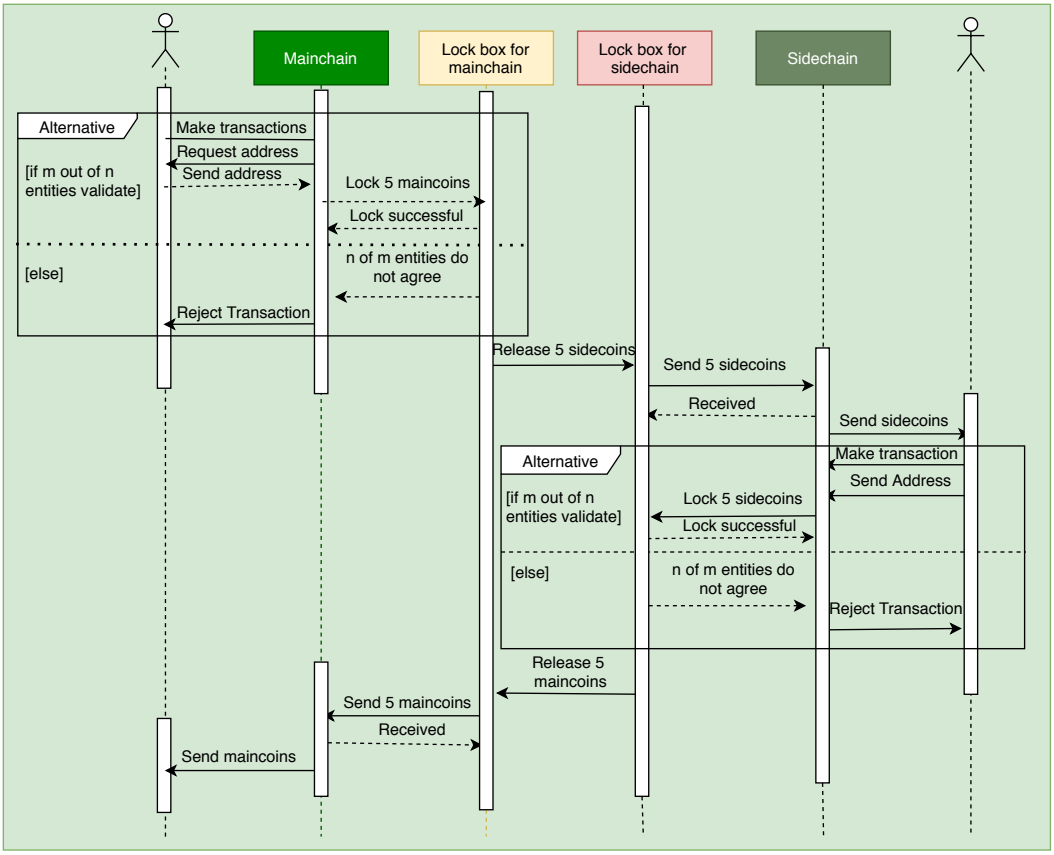


Figure 2.9: The sequence diagram of two-way peg communication

1. The user sends 5 maincoins to the federation that locks the coin for transferring it to the sidechain.

2. The entities of the federation sign the transaction after performing verification. If the certain number of entities approve the transaction, the 5 maincoins are transferred to a user providing address on the sidechain.

3. The user can play rock paper, scissor game with another user using 5 sidecoins and obtains 10 sidecoins if it wins otherwise each user gets 5 sidecoins in case the game is draw.

4. The user sends back 5 sidecoins to the lockbox of the federation. The entities of the federation verify the transactions and transfer the coin back to the mainchain.

2.1.1.7 Performance Metrics of Blockchain Application

Nowadays, diverse kinds of Blockchain-based applications have emerged. Therefore, it is significant to evaluate the performance and success of BC in various use cases and scenarios. Fan et al.[134] conducted a comprehensive survey on Blockchain performance assessment parameters, metrics, and tools. Fan et al. highlighted three tools presented in Table 2.7: Blockbench, Hyperledger Caliper, and DAGbench for analyzing the performance of public and private BC applications under the category of BC benchmark tools and described two simulators: BlockSim and DAGSim. Studies[134, 195–197] presented a set of performance metrics and parameters for assessing DTL (Decentralized Transaction Ledger) and Blockchain leveraged IoT applications as shown in Figure 2.10.

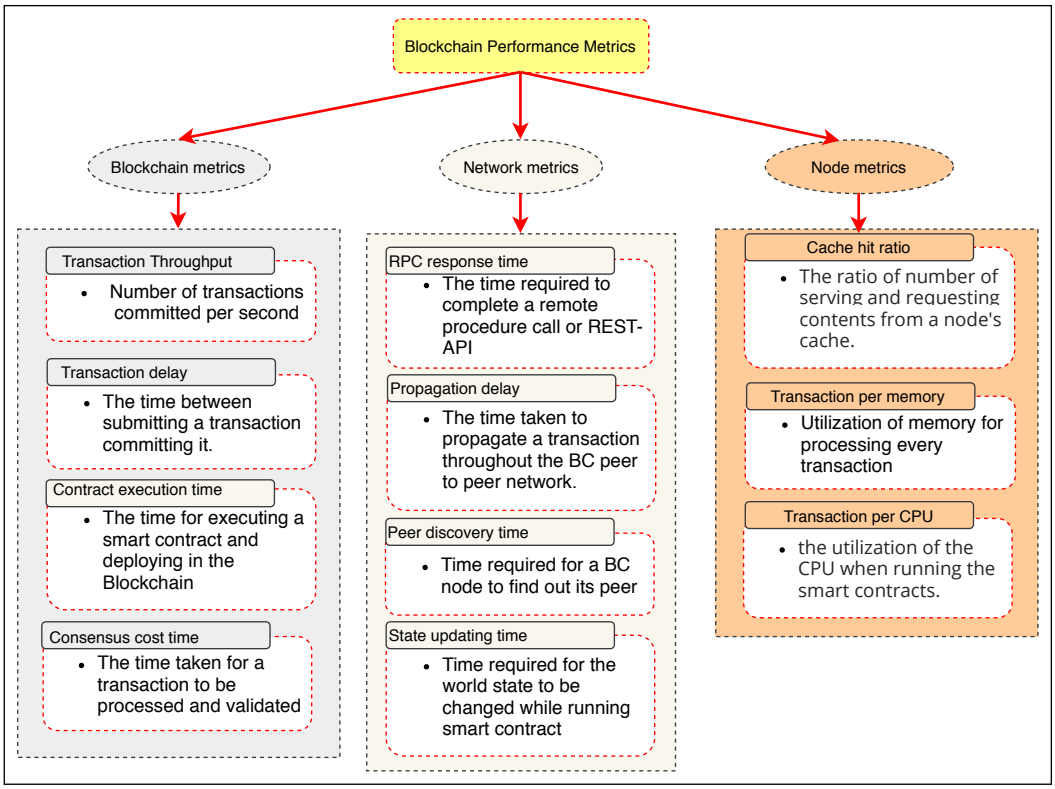


Figure 2.10: The metrics for evaluating BC leveraged applications

Table 2.7: Performance metrics for different BCs

Benchmark performance analysis		
Tools	Performance Metrics	Supported Blockchain
Blockbench[198]	throughput, latency, scalability and fault-tolerance.	Ethereum, Parity[199], HLF[200] and Quorum[201].
Hyperledger Caliper[202]	TPS (Transactions Per Second), transaction latency, resource utilization (CPU, RAM, network, and IO).	Hyperledger Fabric, Sawtooth[203], Iroha[204], Burrow[205] and Besu.
DAGbench[206]	throughput, latency, scalability, success indicator, resource consumption, transaction data size and transaction fee.	IOTA[207], Nano, Byteball[208]
Blockchain simulator		
BlockSim[209], BlockSIM[210]	Block creation rate, system stability and transaction throughput (TPS)	Any private Blockchain comparison with Bitcoin, Ethereum
DAGsim[211]	Transactions arrival rate	IOTA Tangle[212]

2.1.2 Blockchain and Internet of Things(BCIoT)

The Internet of Things (IoT) links individuals, objects, and goods to provide opportunities for capturing data from embedding sophisticated processors, sensors, and actuators, each transmitting data to a centralized server, often Cloud servers. The IoT analytics tools exploit IoT data to turn them into ideas and practice to influence business processes and contribute to new services. However, security and privacy of the IoT ecosystem are significant concerns which have impeded its deployment on a broader scale. IoT network is often susceptible to security vulnerabilities including Distributed Denial of Service (DDoS), Ransomware and malicious attacks. DDoS refers to an attack where a target such as a central server is bombarded with many simultaneous data requests originated from several compromised computer systems, resulting in a denial of service for targeted network users. Further, as the number of devices joining in an IoT network increases, a bottleneck problem can occur in the existing centralised systems while authenticating, approving, and connecting new nodes within the network.

With the solutions of these IoT problems, Blockchain known as distributed ledger technology (DLT) has emerged a breakthrough technology to potentially address some of the IoT security, privacy, and scalability problems. The distributed ledger in the Blockchain is a tamper-resistant, which removes the need to trust the participating parties. IoT covers a diverse range of applications, including smart cities, smart infrastructure, smart grids, smart transportation, smart home, and smart healthcare systems. Blockchain's deployment in the IoT domain has brought a new Blockchain domain in IoT called BCIoT. With BCIoT paradigm, no single organisation has control over the vast amount of data generated by IoT devices. Further, Blockchain technology en-

ables participants to follow up on past transactions. Therefore, data leakage is rapidly detected and remedied. To ensure integrity has become the key research issue in IoT applications, as IoT source code is stored by internet third parties and telecommunications companies that result in lack of trust among consumers. The applicability of Blockchain in IoT network depends on several factors[213]:

1. Blockchain can resolve privacy and security issue if an IoT application needs a decentralized P2P ecosystem.

2. Blockchain could be a promising secured solution if IoT application requires to maintain payment process for its provided services [214].

3. If IoT applications demand to preserve logs and traceability of sequential transactions, the Blockchain can be one of the most effective solutions.

Nonetheless, there are some key obstacles to be overcome when developing an architecture for IoT devices in conjunction with a Blockchain ledger.

1. One of the key challenges of integrating IoT with Blockchain is how the vast quantities of data produced by many IoT sensors can be handled in on-chain. Furthermore, the Blockchain suffers from potentially lower speeds or high latency when processing transactions.

2. Another key issue is to preserve network privacy and transaction confidentiality: the anonymity of transaction history cannot be granted on public Blockchain. Attackers can discover the identities of users or devices by analysing transaction pattern.

In the next section, we reviewed the literature, focusing on addressing the issue mentioned above with IoT and Blockchain.

2.1.3 Blockchain and Cloud of Things(BCCoT)

With the advancement in digital healthcare, a significant quantity of Electronic Medical Records (EMRs) is being generated and exchanged between health institutes and patients to facilitate data collection and provide QoS for the users. In particular, Cloud computing provides powerful health data exchange services, in which EHRs can be processed remotely on Cloud servers, while patients can access information on their mobile devices. The IoT integrated with Cloud computing promises to deliver treatment on-demand, save medical expenses and enhance the quality of experience.

However, information sharing in Cloud environments is susceptible to the risk of potentially malicious attacks and the lack of trust among Cloud vendors, Cloud-based storage, and users. This not only causes adversaries to the medical service and network degradation but also leads to severe data leakage issues. Blockchain technology with high immutable, stable, and trustworthy features can tackle the challenges raised while sharing health information in Cloud ecosystems[215]. Blockchain can secure data sharing across Cloud IoT enabled healthcare networks, in which Blockchain and Cloud are the key contributors to manage user access and data sharing. In particular, Blockchain's smart contracts can automate controlling and authentication of any entry in the Blockchain, ensuring security and protection for insecure healthcare settings. Blockchain paradigms promote cooperation between patients and healthcare organisations to ensure high data privacy and security. Integration of Blockchain into Cloud computing significantly increases security for storage services in Cloud eHealth. Cloud storage acts as peers in the P2P network under

Blockchain administration. Many researchers suggest that health data can be encrypted and stored in the conventional Cloud storage whilst the hash code generated from metadata will be stored on the Blockchain, which allows traceability of data and quickly detects the risks of altering Cloud data. Blockchain can provide specialised, highly reliable, and productive health care services. Blockchain has the potential to transform clinical services, such as health monitoring, patient diagnosis or medical intervention assessment. Consequently, the use of Blockchain models in the health sector will transform healthcare delivery into better patient service and system security.

In addition, Blockchain can provide advanced security services for smart cities applications. Cloud computing offers powerful computational tools for managing massive data streams from all emerging IoT apps for people to deliver services in real-time. With its high-security features, Blockchain shows its high efficiency in managing smart city operations. The convergence of Blockchain and Cloud computing enables smart city architectures to tackle the issue of security and system performance. Blockchain platform offers smart services such as home surveillance, home management, and device access control in smart home scenarios. In particular, Blockchain can be combined with distributed Cloud computing to make data storage and processing more scalable and efficient amongst IoT devices, homeowners, and external users.

Due to the restricted power and storage resources of IoT devices, vast amounts of data streamed from many devices creates a bottleneck for the current IoT systems, which results in low Quality of Service (QoS)[6]. The most common means of storing and processing data is a central database in many existing systems. The centralized repository suffers from several drawbacks:

1. Since a single server is meant to deal with all kinds of customers' queries, customers would not be able to access services during the period of failure[216].

2. There is a risk of violating the data owner's privacy because unencrypted data might be exposed to unauthorized individuals by the entity that administers the centralized storage medium[217].

3. The database can be changed from the server-side without data owner having control or knowledge of the changes in the database [5, 218].

Meanwhile, Cloud computing has virtually unlimited storage and computational resources that can deliver on-demand, reliable, and secure IoT services. The integration of Cloud computing with IoT and BC opens up a new paradigm named BCCoT, which will transfer applications' operation into a safe environment. Indeed, IoT frameworks greatly benefit from the abundance of resources available on the Cloud. At the same time, Cloud can be an additional prominence for real-life applications because of being merged with IoT ecosystems. Additionally, Cloud of Things can transform the current IoT system into a system with minimal managerial effort, high efficiency and quality of service. Cloud analytical tools can support a variety of IoT operations, including historical data processing, information storage and statistical analysis. Cloud data management is used to support end-users to improve IoT services and fulfil customer requirements. Various research identified several key features of Cloud computing, including on-demand support, high processing capacities, automatic management, ubiquitous communication and scalability to support multiple IoT applications. These properties of Cloud computing have motivated researcher to devise diverse kinds of a framework that combine Blockchain, IoT and Cloud of Things technology.

2.1.4 Blockchain and Fog of Things(BCFoT)

Cloud computing alone finds it challenging to handle the flood of information with the proliferation of IoT devices and their constant interactions. Although the Cloud allows users' access to storage, processing and networking resources in cost-effective ways, these centralised services can cause delays and performance problems for IoT devices that are far from the Cloud data centre. Fog computing has emerged on the Internet of Everything (IoE) to reduce energy consumption for IoT devices and significantly increase the processing time of the client's services[219]. The term Edge computing and Fog computing are often synonymous, and both Edge and Fog computing have almost similar features. IoT devices in Fog computing are usually linked to Fog devices via wired or wireless media using Zigbee or LoRa protocol. Both the Edge and Fog computing systems bring facilitates of data processing closer to the data source, and data does not need to be sent to a remote Cloud or other centralised processing systems. Consequently, this technology can reduce the amount of data uploaded to the remote Cloud servers. This decreases the distance required for forwarding data and improves response time for the services, especially for a remote mission-critical application.

The Fog devices geographically spread across heterogeneous networks. Fog computing is a distributed platform that raises the challenges of guaranteeing privacy and security for the Fog devices and their affiliated IoT devices. Fog computing entails a mesh network in which all nodes have almost equal storage and network resource capacities. Fog devices require mutual trust and protection along with the facilities of distributed computing as Fog devices are owned and managed by diverse entities[190]. Therefore, a technology like Blockchain is required to maintain trust in a distributed Fog network where participants don't need to trust one another. Basically, Blockchains eliminate the need for an independent third party and can be undertaken in highly decentralised environments, where all parties including IoT devices, Edge/Fog and Cloud servers need a high degree of autonomy during operation[190].

The distributed feature of Blockchain technology has accelerated its adoption in Fog computing to introduce BCFoT paradigm. However, a full-featured Blockchain cannot be implemented on the Fog nodes due to their restricted storage and computing resources, wide distribution, heterogeneous network, and nodes' selfish behaviour[220]. Further, Not all BC consensus mechanisms are suitable in a Fog ecosystem due to their limited resources. For instance, Proof of Work (PoW) that solves a complex mathematical puzzle requiring massive computational capacity and power, is not appropriate for Fog miners[221]. However, several other protocols such as Proof of Stake (PoS), Practical Byzantine Fault Tolerance(pBFT) consensus are suggested for the Fog network.

2.1.4.1 SDN and Blockchain Technology

Software-Defined Network (SDN) differs from the traditional network in several ways. For instance, unlike conventional network, routing decisions in SDN are made remotely on a controller instead of on each router. SDN[222] isolates network control functions from the forwarding functions so that the network can be dynamic, structured, and programmatically configured to improve its performance and monitoring. Control functions may include the flow control to the switch/routers, routing decision of data packet, governing how router/switch that constitutes forwarding plane handles traffic. SDN manages and orchestrate physical networking tools, including switches, routers, etc., and transfers decision-making to a virtual network control plane. The SDN architecture depicted in Figure 2.11 comprises three planes: 1) Application, 2) Control and 3) Data plane. According to Figure 2.11, the upper layer is called an application plane that supports

end-user with a range of services including mobility, routing, traffic management, network virtualization, and network security protocols developed by various third parties. The SDN application plane executes these services remotely and concurrently. The control plane resides in the middle layer of the SDN architecture that contains the SDN controller installed in the NOS (Network Operating System). The control plane realizes network policies, manages a global database of node placement, information regarding application requirements, and the data flow path of the complete network. Also, the control plane can create virtual instances of the physical controller to meet the maximum number of requests in a limited time without sacrificing the QoS. The lower layer in Figure 2.11 is called a data plane that refers to the physical entities, including switches, routers, base station, and roadside units (RSU) etc. Network devices in this layer receive information from the SDN controller regarding where to transfer the data. Network administrators can use OpenFlow protocol to manage the actions of virtual and physical switches at the data plane.

- **Northbound Interface:** The application plane communicates with the SDN controller about what resources the applications need, and where to send data via a northbound interface which are often RESTful APIs. The control plane orchestrates available network resources and applies its intelligence to discover the optimal forwarding path for the application with respect to latency and security. The SDN controller can also automatically ensure that the traffic for an application is routed according to the network administrator's policies.

- **Southbound Interface:** The SDN controller uses the southbound interface to tell the network infrastructure such as routers and switches how these devices are moving application data. The routing tables that were stored on the devices in the traditional network no longer specify the data forwarding path. Instead, the SDN controller takes an optimized decision about the data path and instructs the router/switches to route data in compliance with the decision of the controller.

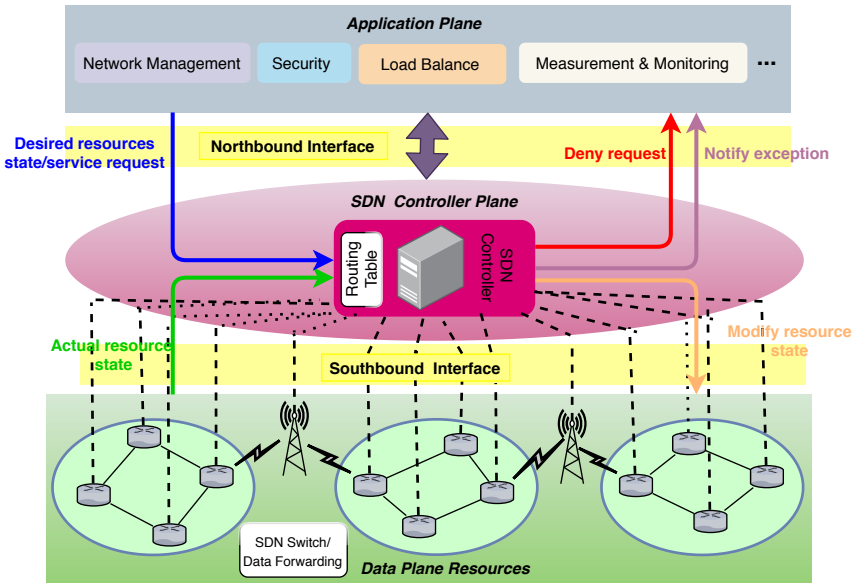


Figure 2.11: The SDN controller as a feedback node

The key concept behind the SDN technology is to separate the controlling functions from the network devices, and a centralized SDN controller manages network functions. This centralized

SDN controller is vulnerable to various cyberattacks including DoS, and single point of failure attacks[223]. To tackle these issues with SDN technology, many recent studies suggest a decentralized SDN controller. However,a decentralized SDN controller raises some issues including the problem of maintaining state consistency among multiple SDN controllers, static flow control between the SDN controller and forwarding plane, which causes a non-uniform distribution of loads between the replicated SDN controllers[224]. Recently, researchers [224, 225] have sought Blockchain technology to integrate into decentralized SDN IoT framework to ensure uniform state among the instances of the SDN controller.

Figure 2.12 describes a modified IoT-SDN infrastructure inspired by Sharma et al. [224, 225]. The proposal incorporates an SDN controller for every infrastructure providers' network and maintains multi-chain. The lightweight multi-chain enabled decentralized SDN-IoT architecture depicted in Figure 2.12 was suggested to address the issues of the current SDN based IoT architecture.

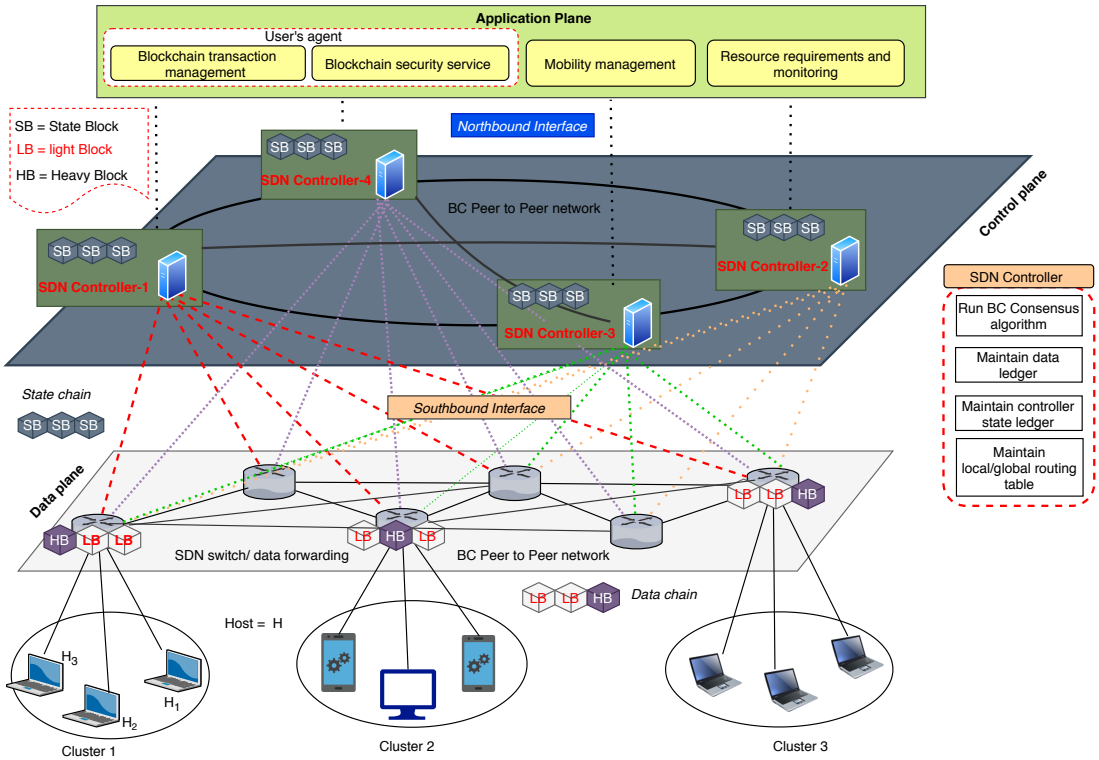


Figure 2.12: The BC enabled decentralized SDN architecture for IoT

Figure 2.12 shows that three clusters of host ($cluster_1$, $cluster_2$, $cluster_3$) contribute computing and storage resources for user's applications. The hosts are grouped, and each group is labelled as a cluster. A cluster of hosts is connected to a nearby SDN forwarding device such as router or switch. The data forwarding devices form a peer to peer network to facilitate Blockchain. In the control plane, multiple SDN controllers are installed where SDN controllers also form a peer to peer network to host a Blockchain. Each SDN controller has full control over the complete forwarding plane like a centralized SDN controller. However, unlike a centralized SDN controller, SDN controllers in Figure 2.12 replicated among multiple servers are connected using Blockchain technologies. Every server hosting SDN controllers executes consensus algorithm, stores distributed ledgers containing routing table for moving data from one host to another host.

The ledger that SDN controllers contain is called state ledger that saves SDN controller operations related information on BC in a linked list fashion after a specific time interval. As a result, the operation of an SDN controller can be resumed while it is down due to cyberattacks. Further, state ledger ensures the same state and integrity amongst the SDN controller's replication.

The other kind of ledger called a Data ledger that is maintained by forwarding devices to store the data generated by their affiliated hosts. A Data ledger contains two kinds of Block: Heavy Block (HB), and Light Block (LB). HB includes data and the hash value of the data, whereas LB contains only the pointer/hash value of the Data. A forwarding device holds the LB for the data produced from its associated clusters and LB for other clusters. This process can provide users with better security than that of the approach to store all data Blocks in a centralized server and hash value/pointer of data Block on the chain.

2.2 BC State-of-the-Art Applications in IoT Field

Blockchain and IoT, Blockchain and Healthcare, Blockchain and Fog computing, Blockchain and Cloud computing, Blockchain and Agent etc., are the keywords that were used to scan literature. The journal and conference papers were downloaded from reputed databases and publishers, including IEEE Xplore, Elsevier, ACM, MDP, SAGE etc.

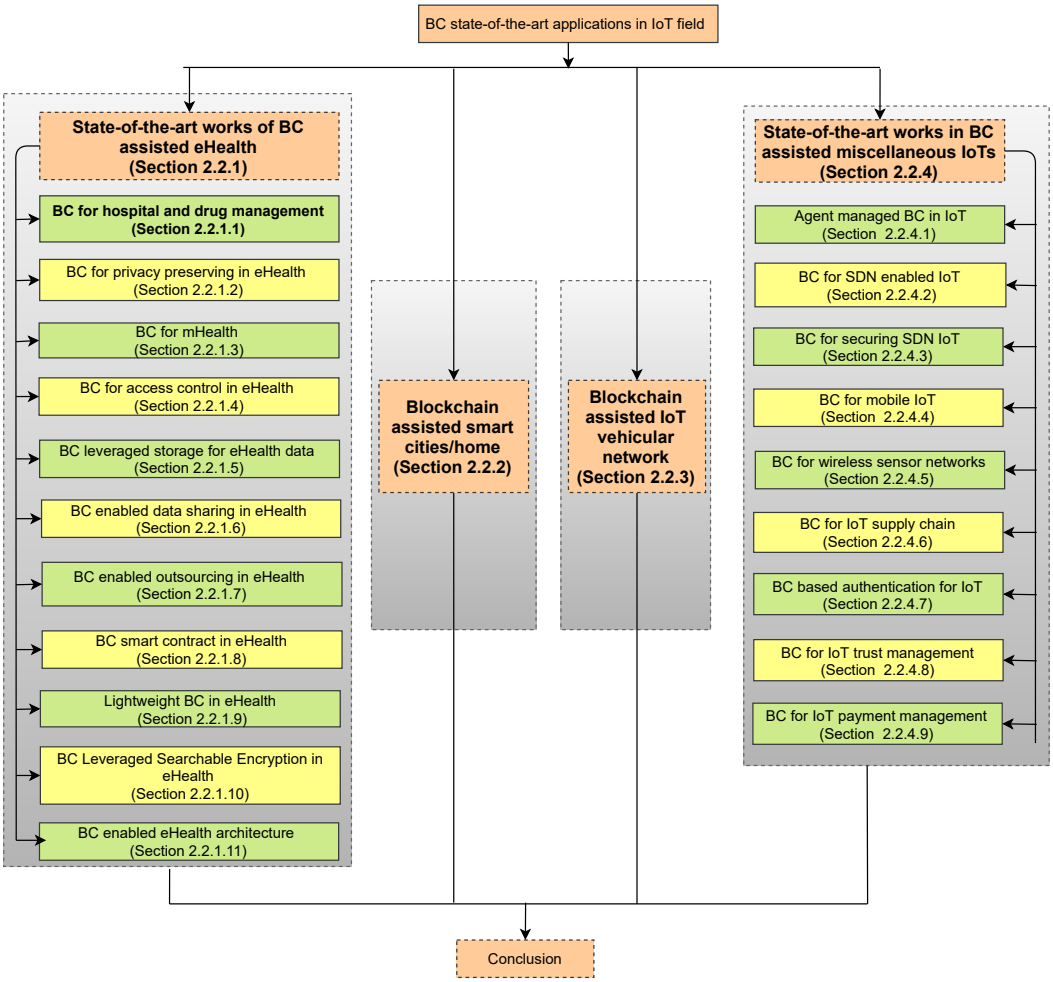
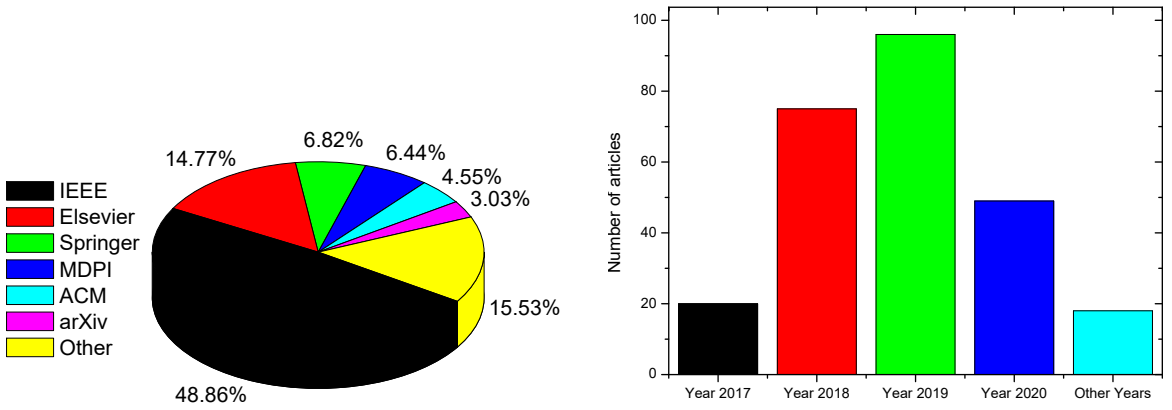


Figure 2.13: The flow diagram of the reviewed literature

Figure 2.13 depicts the flow diagram of the literature reviewed in this article. In Figure 2.14, we presented statistics of the papers reviewed throughout the article. The graph depicted in Figure 2.14(a) shows that the largest percentages of research papers have been retrieved from IEEE Xplore, while the second-highest percentages of publications have been collected from various Journals of Elsevier publisher.

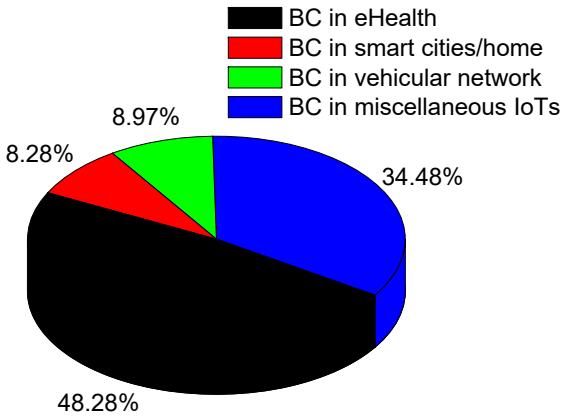


(a) The statistics of IoT literature with respect to publisher

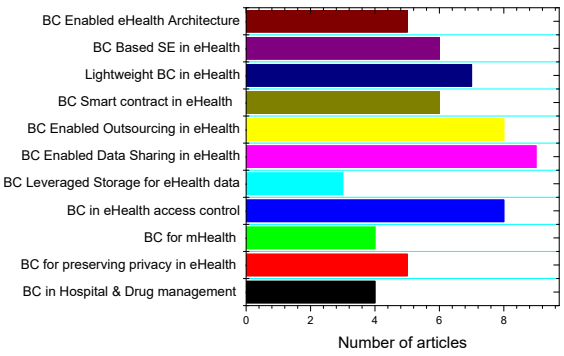
(b) The statistics of IoT literature with respect to years

Figure 2.14: The statistics of state-of-the-art works in BC for IoT

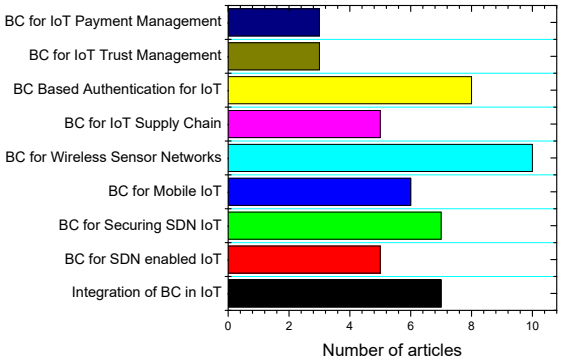
The graph in Figure 2.14(b) displays the reviewed papers with respect to their publication year. We aimed to include the recent existing works on Blockchain and IoT, which are reflected in the chart. The majority of studies included were published in 2019. The statistics of reviewed articles according to applied fields and the BC roles in various IoT applications were shown in Figure 2.15(a), (b), and (c) where the majority of articles are from BC eHealth and the second-highest numbers of articles covers miscellaneous IoT applications. The chart depicted in Figure 2.15(b)(c) shows the breakdown of BC's role in eHealth and miscellaneous IoT applications, respectively. The articles from each major section are synthesized in different tables. The acronym used in summarizing different IoT studies are presented in Table 2.5 and 2.8 respectively.



(a) The statistics of BC's role in different IoT applications



(b) The statistics of BC's role in eHealth



(c) The statistics of BC's role in miscellaneous IoT applications

Figure 2.15: The statistics of reviewed articles according to the role of BC

Table 2.8: The acronym and interpretations-2

Acronym	Explication	Interpretation
SCM	Standard Consensus Mechanism	Common consensus algorithms that are frequently exploited in Blockchain include Proof-of-Work (PoW), Proof-of-Stake (PoS), Delegated Proof-of-Stake (DPoS), Proof-of-Authority (PoA), Practical Byzantine Fault Tolerance (pBFT), and Proof-of-Importance (PoI) etc.
CCM	Customized Consensus Mechanism	Customized consensus mechanism are variations of the standard consensus protocol. The researchers or developers modify the standard consensus protocol to optimize power consumption and increase Blockchain throughput
ECM	Enterprise Consensus Mechanism	The consensus mechanism adopted by the enterprise Blockchain communities such as Hyperledger Fabric, Ethereum
SC	Smart Contract	Smart contracts refer to lines of terms and conditions coded by computer language. Smart contracts are stored on a Blockchain and automatically execute if specified terms and conditions are met.
OfC	Off-Chain	The transactions that are not recorded on the Blockchain and typically stored into centralised databases like banks or other financial intermediaries are referred to as OffChain transactions. IoT data are not usually documented on the distributed ledger rather on traditional databases such as CouchDB, StateDB etc.
OnC	On-Chain	The transactions processed and stored on the distributed ledger in the Blockchain network is called On-Chain. Generally, the pointer or a hash value of the IoT data, financial transactions are recorded on the Blockchain.

2.2.1 State-of-the-art Works of BC Assisted IoT eHealth

In this section, we reviewed state-of-the-art works that are related to eHealth framework. eHealth offers hospital services, and other medical benefits to enable people to rapidly access their health services. Adoption of Blockchain paradigm in eHealth can effectively address critical issues of security and privacy, and increase service efficacy to promote patient care and gradual transformation of existing health system into decentralized eHealth [226] [227–229]. Researchers aim at designing eHealth architecture using IoT, Fog, Cloud, and Blockchain for securely sharing data, managing data storage and network.

Blockchain leveraged healthcare can reform and promote interoperability, authorized access to patient medical records, and secure tracking of prescription, hospitals assets, and wearable sensors during their entire life cycle. The clinician very often requires to access the patient's past disease histories that were created while the patient visited different physicians from different hospitals and clinics. In the most current eHealth settings, patients do not have access to EMRs of the healthcare providers. However, a patient having access to his past histories could avoid the duplication of his medical records, unnecessary medical tests and examination again. The Blockchain can dramatically impact the efficiency in healthcare delivery and costs by providing the patient with full control over his or her past medical records including reports, financial documents, laboratory test result, imaging studies of x-rays, CAD scans, and vital sign measurements. The health data in remote patient monitoring settings is rapidly expanding with other health data, which faces various challenges, including data access, and how data can be accessed outside the healthcare facilities etc. Blockchain provides patients with the ability to boost the authorization and integrity of patient data. We below organized literature according to the role and purposes of BC in IoT eHealth.

2.2.1.1 BC for Hospital and Drug Management

Jamil et al.[159] developed a Blockchain-based vital sign monitoring platform for the hospital facilities. The patients equipped with wearable sensors in the hospital transmit vital signs to the authorized nodes on the BC networks. The architecture was advanced based on a Cloud-driven model with the development of Cloud front-end technologies using HTML5 and JavaScript, to enhance the management of resources within the proposed framework. The BC provided product-centred services using the Representational State Transfer Application Programming Interfaces (REST API), which are either triggered by IoT devices or a web client. A smart contract supported controlled access to the BC ledger to ensure that patient vital sign information is confidential and consistent with data and hosted BC ledger functions across the proposed network. Further, the access control policy was implemented to allow system participants and users to access authorized content and transactions that is, only a doctor may access and manipulate the IoT device. The nodes on the BC P2P network installed couch database to hold the vital sign transactions. A benchmark tool known as Hyperledger Caliper[230] was utilized to evaluate the system's performance in terms of several metrics, including TRL, TRT, TL, and RT. Celesti et al. [231] also proposed an eHealth system that connected the Clouds of a federated hospital using Ethereum Blockchain to build a telemedical laboratory. Although the authors described the healthcare workflow for the proposed system, the extensive performance analysis has not been carried out to demonstrate the feasibility of the system.

The malpractice of healthcare professionals with patients persists in many countries due to a lack of adequate national policies and regulations. Malpractices in the healthcare system include the following: some doctors force patients to perform their diagnosis and medical test and purchases medicines in/from physician's preferred clinics or hospitals. The healthcare professionals

manage patient's health data, and medical tests under their oversight and control using an electronic format where patients are not allowed to access those documents. Consequently, patients need to perform the same test twice when they switch to different physicians. To tackle these issues raised in the traditional healthcare system, Rathee et al.[232] proposed a Blockchain-based hybrid system for processing multimedia produced from IoT healthcare. The Blockchain network applied in the framework consists of two types of nodes such as authenticating nodes or miner nodes and executing nodes. The role of the executing node is to scrutinize whether the transactions that miners accumulate in the Block is legitimate or not. The proposed scheme was simulated using NS2 to analysis its security strength.

BC has an enormous potential to secure the pharmaceutical supply chain. BC can provide an integrated solution for avoiding counterfeited drugs by making the entire drug distribution network traceable to all stakeholders at any point in the supply chain. Haq et al.[233] adopted BC in a drug delivery system to prevent counterfeit drugs. In this system, every transaction generated from drug production to distribution was recorded in a permissioned BC where only trusted authorities can join. As a result, the system can guarantee transparency and facilitates traceability while trading drugs.

Nguyen et al.[234] presented a conceptual, clinical assessment and control framework by integrating Blockchain, Cloud and IoT. They combined the data management system with a data-sharing platform using a decentralized mobile Blockchain network. Data integrity and privacy are ensured using smart access based authentication approach on the access control layer. However, the demerit of the article is that scalability and communication cost issue of the Blockchain has been not investigated.

There has been a tremendous interest among researchers to endeavour Blockchain technologies to provide secure and stable data storage for healthcare. However, a few countries, including Estonia, Peru, have already adopted Blockchain health management in practice. In Peru, a Blockchain-based private health purchase management system[163] was recently introduced. The Blockchain was implemented in the Amazon Cloud to control the medical supply chain, ensuring secure communication between the sale managers, manufacturers and clients. The smart contracts are developed for storing medical sensor data to prevent data from malicious alternations or modifications. The drawback of the proposed scheme is that data confidentiality has not been addressed. Kang et al.[235] highlighted the effectiveness of Blockchain paradigm in providing health services through IoT and Cloud of Things. However, performance assessment for the proposed scheme has not been carried out.

2.2.1.2 BC for Privacy Preserving in eHealth

Preserving privacy in eHealth system can make contact efficient between physician and patient, which is crucial for quality treatment, improved autonomy and tackling economic damage, embarrassment and discrimination[236].

Researchers[164–166, 237] designed BC-based IoT eHealth to ensure patients and healthcare providers privacy. The work in[164] is a privacy-preserving health data exchange approach that integrated IoT network and Cloud storage. The conceptual model comprises three layers: 1) data collection, 2) data storage and 3) data exchange layer. Electronic medical records (EMRs) are securely stored in the Cloud layer using smart contract technology, while the indexing of the records is maintained in the Blockchain to secure medical records. Consequently, EMRs cannot be inappropriately changed or manipulated. However, a real prototype of the framework is yet to be implemented.

In [165], BC was undertaken to build a privacy-preserved Cloud health data platform. Smart contract regulated encrypted health records are stored on the Cloud BC ledger. The vulnerabilities to data confidentiality are effectively tackled by encrypting data before inserting those into the Blockchain, which improves transparency and security of Cloud data storage. The limitation of the work is that comparisons have not been made between smart contract-based schemes and traditional schemes, and the model was not implemented to analyze performances. Similar work in [166] advanced a stable Cloud-based Blockchain EHR platform with four entities: 1) a key generation centre, 2) healthcare professionals, 3) Cloud patients, and 4) data customers such as insurance firms. The time-stamped medical data is stored in the Blockchain, which increases the validity and traceability of health records. The weakness of the article is that a smart contract for managing data storage has not been implemented. The BC ledger is transparent to all the entities on the BC peer to peer network. Miners verify the contents of the Block before writing it in the distributed BC ledger. This openness of BC is a major threat to the privacy of patients in the eHealth system. To fix this issue, Rahulamathavan et al.[237] restructured the BC peer-to-peer network to adapt the techniques for Attribute-Based Encryption (ABE). The authors classified BC nodes as cluster head, attribute authorities, and miners based on their roles in the BC network. The cluster head of the BC network is connected to IoT devices for collecting IoT data. The cluster head performs computationally intensive operations, including processing and encryption/decryption for the data while Attribute Authorities(AA) is responsible for providing doctor, nurse and other healthcare professionals acting as miners with the attributes required to decrypt data. The selected miners can decrypt the Blocks using attributes obtained from AA for verifying and validating Block.

2.2.1.3 BC for mHealth

Mobile devices have enabled healthcare providers to improve patients' engagement and participation in treatment processes using mobile-assisted secure text messaging, patient apps, and telemedicine. In the existing settings, secure messaging between patients and healthcare professionals remain the primary use of mobile devices in IoT healthcare. However, BC related works[69, 160, 238, 239] had incorporated mobile apps to safely capture health data from a patient's wearable sensors and deliver rapid health services to patients.

Liang et al.[238] advanced a Blockchain-enabled mobile Cloud network where data streaming from wearable sensors are transmitted to Cloud server via smartphone. The authors aimed at developing a patient-centric platform to share health data between healthcare providers and insurance farms. The system included six groups of users, including consumers, wearable devices, healthcare providers, insurance companies, the Cloud ecosystem and the Blockchain network. The Hyperledger Fabrics, which is an enterprise permissioned Blockchain, was utilized to validate and preserve the patient's data while sharing the data with different stakeholders. Blockchain was deployed in the Cloud to serve three purposes:

1. To ensure the integrity of data entry.
2. To process requests from external sources to gain data access.
3. To implement access control for user verification.

The Cloud server is configured to connect to the participants on the peer-to-peer distributed Blockchain network using a Hyperledger Fabric client that protects the anonymity of the Cloud user's requests. However, they overlooked security issues such as malicious attacks of IoT devices.

Meanwhile, Nguyen et al.[160] projected a mobile Cloud Blockchain network that was designed to integrate various EHR systems to share health data between health care providers and patients. Blockchain was embedded in Cloud server where smart contracts handle user transactions for data access. In the Cloud, decentralised storage system(IPFS) made data sharing more effective as opposed to centralised distributed systems in terms of low latency and privacy. The system has enabled IoT users such as doctors or patients, to securely exchange data using their portable devices, including smartphone, laptop. A private Ethereum Blockchain network implemented on the Amazon Cloud was used to analyze the performance of the proposed scheme. Ni et al.[69] figured out HealChain, which is a mobile healthcare system that comprised of three layers such as data collection, verification and storage layer. The research limits the number of Blockchain participants to keep mining cost moderate. Further, they developed an optimal decision-making process to maximize the economic benefit of carrying out mining tasks. However, the authors did not describe which kinds of simulation tools or programming language was used to conduct performance analysis.

Ichikawa et al.[239] presented a Blockchain leveraged mHealth framework to immune health data from tampering. They developed a mobile App using JavaScript Object Notation format to collect data from wearable sensors and store those in a private Blockchain of Hyperledger Fabrics. The authors examined the successful inclusion of health data in the fault networks. However, the security issues between sensors, mobile App have not been addressed.

2.2.1.4 BC Leveraged Access Control in eHealth

The security of eHealth systems is a vital issue because a security breach can endanger a patient's privacy, health, or even his or her life by maliciously altering diagnostic data[240]. As one of the key security features, access control ensures that only authorised users with correct privileges can access health services. Access control refers to a user, group, or organization's rights to access health information within the domain. Naturally, health systems need to implement fine-grained access to control[241]. For instance, only previously registered healthcare providers should be given access to an Electrocardiograph (EKG) in a real-time monitoring service. Several approaches[40, 161, 242, 243] have been proposed in the Blockchain eHealth system to solve the problems pertaining to authentication and access control.

Focusing these issues, Tanwar et al.[40] suggested Blockchain ledger to store access policies of the medical record. The authors designed several algorithms that defined access policies for healthcare providers on their patient-centric healthcare framework. They analyzed the performance of the system in terms of throughput and latency using different tools, including Hyperledger Fabric, Composer, Docker Container, Hyperledger Caliper, and the Wireshark capture engine.

Wang et al.[161] targeted a data-sharing system with fine-grained access control to better protect the privacy and accessibility of health data. Wang et al.[161] designed a decentralised Cloud architecture that incorporated an interplanetary file system (IPFS) for making decentralised storage, an Ethereum Blockchain and an ABE(Attribute-Based Encryption) platform. In this work, a smart contract-based access control management system has also been suggested to conduct keyword searches in the decentralised Cloud storage, which enhances the QoS(Quality of Services) and privacy of the framework. However, the drawbacks of the research is that data security and the delay arising from ABE and access control approach were not analyzed. Wang et al.[244] also sought the Blockchain Cloud infrastructure to store medical data using a Blockchain-enabled the authentication approach of medical data transactions. Any modification to the Cloud records is detected through the Blockchain's P2P network. In fact, this model eliminates the costs of manag-

ing data storage by third parties. However, Blockchain prototype has not been implemented in the article.

Islam et al. in[242] presented a framework to assist health prescriptions (HPA) so that patient receives the recommendation from the physicians. The system provides IoT devices with a security access token (SAT) upon successful authentication, which defines the privileges of medical IoT devices and their services or resources for the user. The IoT devices include the encrypted SAT while asking services from the system. The proposal also includes an access control mechanism based on the OpenID to prevent unauthorized access to medical devices. However, the model is conceptual, and no performance analysis was carried out.

Ramani et al.[243] presented a medical data accessibility approach on the Blockchain. The system allows healthcare professions to append and retrieve health data with the consents of a patient. A private Blockchain was considered to analyze security theoretically. However, the authors did not evaluate the performance of the proposed technique using any simulator or building prototype.

GDPR health regulations outlined privacy laws across Europe to protect user's control and confidentiality on his or her health data. According to the regulations, a service provider must incorporate facilities of the user's consent and the withdrawal of that consents in their system. The service provider must generate a report at the request of the user on how the user's data is being processed and used. Further, the service provider must also provide the customer with all the data in a format that is readable on the computer. The research in [53] presented a conceptual eHealth framework by augmenting Blockchain technology and Cloud to share health data with the authorized user in an efficient, transparent manner and maintaining compliance with data regulations such as GDPR. The authors inspected the quality of the health data using a machine learning technique to ensure the QoS of the shared data. The limitation of the paper is that performance analysis has not been done.

2.2.1.5 BC Leveraged Storage for eHealth Data

Medical data is typically processed and stored in the Cloud servers under the administration of different Cloud Service Providers (CSP) in traditional Cloud IoT-enabled healthcare systems. CSP should be transparent but vigilant about the risk of leaking out sensitive patient information. EHRs(Electronic Health Records) are also susceptible to various forms of data-storage attacks while using Cloud security tools. BC can be an automated technological solution to make the current storage method for health data more secure and effective. BC can preserve the integrity of data while ensuring tamper-proof. One way to store records with a Blockchain is on-chain storage. But, BC demands high cost to insert a Block on-chain[245]. On-chain storage is considered neither financially nor technologically feasible. However, another data storage method called off-chain can be implemented on the BC network. In an off-chain method, the hash code of a piece of data, which is relatively small, is stored in the BC ledger and the data is stored in traditional repositories. The storage cost in the on-chain method is low because the size of the hash value is relatively small[246]. Most research presented in Table 2.9, and 2.10 has been endeavoured to address the storage cost issue on the BC network by following the off-chain database approach. Zheng et al. [53] outlined a conceptual model for continuously sharing personal health data using Blockchain-based decentralised Cloud storage. Health datasets are usually encrypted and stored off-chain in conventional Cloud storage, whereas only hash values of the data are inserted into the Blockchain to resolve the storage burden in the Blockchain framework. However, the real prototype of the framework is yet to be developed.

2.2.1.6 BC Enabled Data Sharing in eHealth

The protection of patient's privacy is a significant issue due to the sensitive nature of the medical data while exchanging EHRs. BC has appeared as a potential solution to this issue because of its decentralisation and manipulation resistance characteristics[247]. In [75], Xia et al. schemed a medical data sharing model called MeDShare which utilized Blockchain for exchanging data amongst untrusted Cloud Service Providers (CSP). The researchers devised an access management architecture that exploited smart contracts to track access behaviours of data users and detect data breaches. The Blockchain-based CSPs could enable auditing and ensure healthcare professionals' provenance without compromising the confidentiality of data. However, concerns associated with access control of confidential data are needed to be efficiently resolved in the Cloud-based data processing. To address this issue, the research in [248] included a secure cryptographic approach for ensuring efficient access control and user's authentication for transferring data in the Cloud layer.

Physicians are normally specialized in delivering medication and care for a specific illness. However, treatment for many diseases needs cross-border medical knowledge from different medical practitioners worldwide. The BC platform can facilitate the exchange of a healthcare professional's expertise for more precise medical care, personal diagnosis, and treatment. Wang et al.[161] suggested a hybrid healthcare system to combine knowledge from three fields: 1) artificial intelligence, 2) computational experiments, and 3) parallel execution (ACP) to expedite more precision medical care and treatment. Firstly, an artificial healthcare system (AHS), known as "descriptive intelligence" was developed to simulate and model the static and dynamic characteristics of patients and doctors. Secondly, computational experiments were used to integrate different types of disease scenarios to assess and evaluate the applicability of specific therapeutic regimens in AHS. The phase is called "intelligence predictive". Thirdly, the final regimen was chosen from a list recommended by experts and was carried out in parallel, both in the AHS and the current health care system, to provide "prescriptive intelligence". The system deployed a consortium Blockchain that involves patients, hospitals, health officials, healthcare institutions, and medical researchers, and Blockchain-powered smart contracts to allow electronic health records (EHRs) to be exchanged, checked and audited.

Blockchain-based health data management is a transparent and open framework to support better healthcare services. Indeed, the combination of Cloud, IoT, and Blockchain can offer great advancements in smart medical services[249]. In[91], a decentralised Blockchain data security scheme was designed by Dwivedi et al. The infrastructure comprises five components: 1) overlay network, 2) Cloud servers, 3) healthcare providers, 4) smart contracts and 5) patients. In the work, Blockchain was linked to Cloud storage using a P2P network where each Cloud storage holds medical records in the form of Blocks and these Blocks' hash values are stored in the Blockchain which facilitates the tracking of any changes in the Cloud data. A dual encryption scheme is also proposed to safeguard data from potential attacks. The weakness of the article is that actual simulations have not been rendered on the suggested security scheme.

Nguyen et al.[160] advanced a novel EHRs sharing architecture based on Blockchain and shared interplanetary file system storage (IPFS). To enhance the security of EHRs during their exchange, smart contracts was designed to build a trustworthy access control mechanism. In addition, a data exchange protocol was developed to handle user access to the EHRs network. The usability tests were conducted on a mobile Android application, and Amazon Web Services provided Cloud. Results of the assessment indicate that the suggested approach is feasible on different e-health scenarios.

Shen et al.[250] proposed Medchain that is a platform for sharing medical record. The authors leveraged two separate decentralized networks: a BC P2P network and a normal P2P network. The Blockchain network stores data, session, and operation fingerprints, such as immutable data digests, while the normal P2P network stores data and session descriptions, which are mutable. A session for packaging and removal of the mutable information is introduced in the data sharing process, which can reduce overhead storage considerably.

Fan et al.[251] designated a Blockchain-based medical sharing system where the provincial hospitals collect medical summaries from EMRs of the regional hospitals and community centres. The provincial hospitals pack medical data into Block after processing and then transmit the Block to the consensus nodes. Hospitals acting as both orders and endorsers play the role of initiating queries, verifying, and validating Blocks. A hospital can opt to maintain health data in their ledger locally or submit it to the Blockchain.

2.2.1.7 BC Enabled Outsourcing in eHealth

In recent years, outsourcing health services to a Cloud service provider has become significant to reduce the local computation burden[252]. Outsourcing is described as the act of shifting an organization's internal activities or services and decision-making to external suppliers following long-term contracts or agreements[253]. However, outsourcing tasks to a Cloud computing provider bring a few other challenges. The Cloud service provider might be curious on user's sensitive data and breach the client's privacy, the client needs to make contracts with the service provider so that the data privacy cannot be breached[254]. Research [168, 169, 255] has investigated Blockchain as a promising solution to the service outsourcing challenges such as security, privacy, payment and contract.

The authors in [168, 169] presented a Cloud assisted eHealth framework using Blockchain to secure outsourcing EHRs among medical users. An Ethereum Blockchain framework has been utilized for handling user transactions without the need of a trusted entity. The integrity and reliability of EHRs generated by patients and clinicians during the treatment process were guaranteed by inserting medical data into the tamper-proof Ethereum Blockchain in the form of transactions. However, a smart contract for managing service has not been investigated.

The research in [169] envisioned Cloud-based crowdsourcing to develop a medical remediation and evaluation framework called CORUS. Crowdsourcing refers to a process of collecting works, information, or views/opinions from a wide number of people who send their data via the internet, social media, and mobile applications. People interested in crowdsourcing often work as paid freelancers while others can voluntarily perform tasks[256].

Crowdsourcing on the traditional platform is exposed to several shortcomings such as a single point of failure, controller's silent misbehaviour, a conflict of opinions between the task requesters and the workers[255]. Blockchain, a revolutionary decentralised model, can be adapted not only to remove the limitations of the conventional crowdsourcing schemes but also to usher technological advancements including decentralisation and transparency[255, 257]. The decentralized ledger in the Blockchain technology increases the reliability of recorded documents and the efficiency of the proposed crowdsourcing system[169]. Additionally, Park et al.[169] applied Blockchain to attract large numbers of participants by offering an incentive for providing reliable information. The shortcoming of the article is that the performance analysis of the Cloud Blockchain has not been investigated.

2.2.1.8 BC Smart Contract in eHealth

With the emergence of Blockchain, smart contracts have become one of the most sought-after technologies because of their automated nature[258]. A smart contract refers to an agreement and rules encoded by computer programming. Smart contract stored in the public ledger is automatically running on the Blockchain without the need of the third party when the contract associated event is triggered[258].

Daraghmi et al.[170] developed a timed smart contract-based medical record access and permission management architecture. The contracts introduced in the research control transactions and monitor computations on the EMRs through implementing appropriate user's policies. The author suggested an incentive-based mining process to eliminate the need for digital currency. In this mining process, the next Block would be created by the node with the low rating and the nodes with higher rating participate in approving the Blocks on the Blockchain network. This ensures consistency between suppliers and ensures the system's sustainability. The experiment was carried out on Ethereum which is an open-source platform to feature the smart contracts using solidity language. However, the article did not handle the security and privacy issue of storing and accessing continuous health data onto the Blockchain.

Kazmi et al.[175] developed a Blockchain-based remote patient monitoring system where smart contracts were made to enrol patients and healthcare professionals, to provide licence for the wearable sensors and other medical services. The system can generate an emergency alert in real-time, thus promote the consumer and healthcare professional's engagement in remote patient monitoring. The smart contracts for the proposed scheme were written on the Ethereum platform. The Remix which is an open-source web environment was utilized to test, debug and deploy their smart contracts. However, the security and privacy issues while retrieving data from wearable sensors were ignored.

Hang et al.[167] proposed a Blockchain leveraged medical platform to protect the management of EMRs across different hospital departments. The EMR management system utilized smart contracts to store, health data, record logs, regulate the access to medical data among different health organizations. They carried out an experimental test of the framework on a network comprising different hospitals to demonstrate the feasibility of the system in terms of efficiency and efficacy. The smart contracts were designed on Hyperledger. The design and experiments were described in details. The article in [10] discussed how to build e-Healthcare systems and services using Blockchain and IoT technologies.

Malamas et al.[176] reckoned Blockchain technology to design a forensics enabled framework for medical devices. The system includes a fine-grained authorization technique using smart contracts on the Blockchain. The smart contract defines the policies and enforces the integrity and confidentiality of transaction logs. The Proof of Stake, consensus mechanism validates the transactions in the Blockchain.

A wide range of queries from patients, clinicians, healthcare professionals and researchers are usually issued to a biomedical database using suitable application programming interface (API) at any given point in time. In traditional log record system, ensuring tamper-proof of data and user's queries is crucial. Mytis et al.[180] suggested Blockchain guarantee the integrity and non-repudiation of retrieval information from the conventional biomedical database. The system comprises three components: a) a data user front-end interface used by third parties to make queries b) an interface for interacting with biomedical interface c) smart contract in between user-interface and database interface to record all user's queries in the Blockchain. The smart contract was developed on the Ethereum Blockchain using Solidity language. MongoDB database is deployed to

store biomedical data.

2.2.1.9 Lightweight BC in eHealth

Blockchain implementation requires immense computational power thanks to its mathematical principles such as cryptographic key systems, the Merkle Hash Tree and Proof of Work (PoW)[259]. Most importantly, IoT devices are typically inadequate in performance. Researchers[10, 17, 260–263] have proposed a variety of ideas to optimize current BC technology.

Ismail et al.[260] proposed a healthcare architecture using a lightweight Blockchain. The authors geographically divided the Blockchain network and defined different roles of BC nodes. The cluster head called Head Blockchain Manager (HBCM) handles transactions and make Blocks. The HBCM maintains a single copy of ledgers for its members, thus avoiding fork. The customized Blockchain can reduce computational and communication delay but can not guarantee the tamper-proof of the ledger. The proposed scheme was simulated on NS3 and was compared with Bitcoin Blockchain in terms of efficiency and computational cost. Srivastava et al.[261] optimized the power consumption of the BC-based healthcare using lightweight cryptographic techniques such as ARX encryption scheme. The Ring Signatures was used to enhance the privacy properties including the singer's anonymity.

Ray et al.[10] also launched an improved IoT-based eHealthcare Blockchain framework, called IoBHealth where the IoT-based Blockchain network for accessing and managing EHR data in eHealthcare is more robust, secure, open and effective. Attia et al. [262] implemented an IoT-Blockchain healthcare architecture to track patients via connected devices. The authors used Hyperledger fabric as Blockchain and implemented a Graphical User Interface (GUI) that enables a network user to display data ledger in clear visualizations and dashboards. Further, the system adopted Naming Data Networking protocol instead of using device identifiers which allows data mobility between different entities.

In chain structured Blockchain, a Block is propagated throughout the network after a miner completes Proof of Work for the Block. This brings its problem with scalability and high network overhead. The research in [264] advanced a scalable Blockchain for remote patient monitoring by incorporating GHOSTDAG protocol which is transaction confirmation protocol. GOSTDAGE mechanism considers each transaction as a node rather than a single large chain of Blocks.

The research in [17] attempted to address the challenges of integrating Blockchain with wearable sensors. The system includes different entities including the Blockchain network, Cloud storage, healthcare providers, smart contracts and patients equipped with wearable IoT devices for healthcare purposes. Blockchain algorithms are run on a hierarchical topology of network nodes where a node with high computational power is nominated as a cluster head for a group of nodes to examine and process Blocks as a representative of its members. Although this approach might address the problem of poor scalability, traffic overhead and power consumption, the avoidance of global consensus mechanism are vulnerable to cyberattacks and sustainability.

Yang et al.[263] proposed a novel consensus mechanism for executing on the eHealth BC. The proposed consensus protocol was called Proof of Familiarity (PoF) that entails a collaborative medical decision making for offering medical services to a patient. In this process, the system enables a new patient to ask for experience of a cured patient given with their similar symptoms and diseases, the medical verdict from several physicians, and the strategic policies from insurance providers. The feedback from every party including healthcare providers, and previously cured patients are used to constitute a favourable joint medical decision. This decision and the hash of the medical data are stored on-chain, and medical data is stored in a local database on off-chain.

The shortcoming of the paper is that prototype is yet to be implemented to study the feasibility of the proposed consensus mechanism.

2.2.1.10 BC Leveraged Searchable Encryption in eHealth

With the rapid development of Cloud computing, the original storage way of health data has been changed[171]. In general, health data are sensitive and need protection against unauthorised access. Health data is typically encrypted before uploading to the Cloud storage. The efficiency of accessing these data on the Cloud depends on the mechanism of encryption approach[265]. Searchable encryption (SE) which is a promising encryption technique guarantees data security, without compromising data searchability[171]. However, most current such schemes, particularly the searchable public-key encryption schemes (SPE) are vulnerable to the adaptive leakage-exploiting attacks or unable to meet the efficiency requirements of realistic applications[171]. To achieve a secure and efficient keyword search in the healthcare system, researchers have suggested merging Blockchain technology with a traditional Cloud storage system.

Chen et al.[171] advanced searchable encryption supported healthcare framework using Blockchain technology. The system saves the search index on the Blockchain while the data is stored in the public Cloud. The consumers require to obtain permission and encryption key from the owner to access the data. The system utilized the complex Boolean expression to extract the index-building EHRs and supported complex queries that allow different healthcare agents to request permission to access and interact with the medical records, which differs from the previous studies in [266]. Smart contract on the Ethereum Blockchain was designed to trace monetary rewards, including transaction fees, in multi-user setting between the parties involved.

Wang et al.[267] contrived a Cloud assisted consortium Blockchain-based framework for storing and sharing electronic health data. The Blockchain stores encrypted keywords for facilitating the quick search of health data uploaded in the Cloud. They defined the structure of Blocks and transactions and implemented primitive cryptographic protocols to store data securely. The Cloud database support re-encryption of the ciphertext and sends the re-encrypted ciphertext to the specified data requester when the patient has agreed with the data owner. The authors[268] also presented a Blockchain assisted searchable EHR storage system. The Cloud server stores the health data using attribute-based encryption to ensure fine-grained access control of EHRs. The Blockchain stores keywords of the EHR data, which is used to build indexes to enable data visitors to find data content on the Cloud storage. Noh et al.[269] recommended Blockchain to record access logs of medical record managed by Cloud service providers. The paper also included a proxy re-encryption scheme for securely sharing patient data.

2.2.1.11 BC Enabled eHealth Architecture

Fog computing has many benefits and is suited for applications that require fast response time, low latency, and real-time processing, for specially healthcare[270] [271]. However, the Fog computing brings concerns regarding heterogeneous platform, security, privacy, trust, and resource management [272]. To answer these issues, Blockchain technology has been adapted in Fog enabled healthcare system. In the context of video stream processing, Islam et al.[271] uploads the data to the Fog server deployed within the vicinity of the video camera instead of Cloud. The authors developed a human activity recognition platform that included Blockchain-based Fog-Cloud computing. They identified important features from video stream before applying data to a multiclass SVM classifier with error-correction out code framework. The strength of the paper is to analyse

the accuracy of the activity recognition system using different datasets. However, the authors did not describe how Blockchain has been utilized in the proposed framework and no performance analysis has been conducted regarding Blockchain, Fog and Cloud platform.

Akkaoui et al.[162] proposed a hybrid Edge Blockchain-based eHealth architecture. The architecture consists of four layers: 1) end-user 2) Edge pool 3) global Blockchain 4) off-chain storage. The idea is like the work in [270] in the context of running mining process on the Edge pool to increase the throughput and transactions processing latency. The Edge pool consists of several Edge devices to check the validity of the transactions and classify the data as normal or abnormal. The global Blockchain Ethereum stores the Block containing metadata of EMR and body area sensors data whereas [270] suggested to run the mining process on the Edge networks and store the Block containing metadata on the Edge network. [162] used an extra global Blockchain that can increase the latency of processing Blocks. The authors [162] also developed several smart contracts to establish role-based access to patient data.

2.2.1.12 BC for Tackling COVID-19 Pandemic

Quarantine has been an effective strategy to stop the spread of COVID-19 since the beginning of the pandemic. Current quarantine activities include social isolation, surveillance, and tracking of COVID-19 infected patients. However, the virus's propagation is too fast for manual and often ineffective human contact tracing systems to contain the virus. Bandara et al[273] implemented a Blockchain assisted automated digital contact tracing system which records positive COVID-19 cases to notify individuals in their local vicinity without revealing sensitive personal information. The authors developed a wallet on the Android smartphone to store self-sovereign identity (SSI) proofs as user's digital IDs and activity tracing data on the Blockchain network. They also developed a machine learning model to detect anomalies from the activity tracing data on the Blockchain.

To automate contact tracing and maintain privacy, other researchers deployed a decentralized Blockchain to ensure transparency and immutability of the tracing data. Pandey et al.[274] presented CovidBloc, a Blockchain based COVID-19 cases tracing system. CovidBloc involves three components: 1) a smartphone app 2) a dashboard for healthcare professionals 3) a Hyperledger Fabric Blockchain as a backend server to store contact tracking data.

The outbreak of 2019 coronavirus pandemic (COVID-19) underlined the urgent need for robust smartphone-based contact tracing solutions to halt the infection from spreading further. However, due to the nature of contact tracing, public concern over privacy issues and bottleneck problems have impeded the widespread adoption of standard contact tracing apps. Xu et al. [275] designed a BeepTrace which is a Blockchain assisted privacy-preserving contract tracing system. The system utilized Blockchain to desensitize the ID and location information of COVID-19 infected individuals. BeepTrace maintains two-chains for preserving user's privacy: geodata tracing chain and passive notification chain. The authorized solvers have access to the tracing chain and can discover contact matching using desensitized personal location information. The notification chain will provide match results (just a pseudonym or fingerprint) enabling vulnerable users to self-match locally.

Detection of COVID-19 infected individuals becomes difficult when they internationally travel. However, cutting-edge technologies including Blockchain, and artificial intelligence have emerged as potential solutions for tracking individuals infected with COVID-19. The smart contract can store information about early COVID-19 case identification on the Blockchain while also protecting individual privacy and data security. Rimsan et al.[276] suggested a Blockchain based COVID-

19 case detection system. The framework that can aid in making key decision about COVID-19 prediction and prevention will be validated using design science research (DSR) techniques. Azim et al.[277] investigated the utilization of Blockchain for managing the pandemic data that assures consistent and reliable data storage across various nodes in order to track down COVID-19 cases.

Since the COVID-19 vaccine was discovered and made available to individuals, the distribution and administration of the vaccine has become increasingly challenging. Due to its vulnerability to a single point of failure, the current system is unable to provide complete transparency, traceability, immutability, and trustworthiness for vaccine delivery and distribution. Musamih et al.[276] designed a smart contract on the Ethereum Blockchain to manage COVID-19 vaccine related data including distribution and delivery information. The smart contract was constructed to automate the delivery and distribution of COVID-19 vaccines and generate events and notifications in the event of violations to the supply chain process of the vaccine. To handle the requirements of large-scale data storage, the system utilized off-chain storage such as IPFS, Google and Amazon storage to connect Ethereum Blockchain.

Guaranteeing immutability of vaccination records and secure access to the COVID-19 information for hard immunity against COVID-19 is a burning question. Recently uncontrolled sharing of misinformation about the COVID-19 vaccination has prompted researchers to search for a dependable and reliable alternative to existing digital systems. To address this problem, Deka et al.[278] proposed a Blockchain leveraged method for recording and tracking COVID-19 vaccination and ensuring proof of immunity for individuals. The approach utilized a smart contract to record vaccination information on the Ethereum Blockchain.

Due to a lack of effective data management and information exchange, SARS-CoV2 or COVID-19 has been spreading rapidly over the worldwide, with a rising death toll and infection rates. The existing database system is largely centralized and regulated by third parties. Data tampering is a threat to the traditional database system. The ”Blockchain” is a distributed shared ledger system that maintains multiple copies of synchronizing and verifying databases on a peer-to-peer network.

Sharma et al. [16] outlined a Blockchain leveraged architecture to combat COVID-19 pandemic. The authors discussed major Blockchain based applications such as vaccine registration, contract tracing, and infection detection results recorded through IPFS to address COVID-19. To combat this pandemic crisis, BC technologies can improve medical supply chain solutions, facilitate decentralized outbreak tracking, preserve privacy, and provide secure day-to-day operations. The Blockchain in conjunction with artificial intelligence, big data, and cloud computing can provide viable solutions to COVID-19 pandemics[16].

The Blockchain-based healthcare studies covered in this paper is briefly described in Table 2.9, 2.10,2.11, 2.12 and 2.13 with respect to diverse attributes. The acronym used in analyzing literature is explained in Table 2.5 and 2.8 respectively.

Table 2.9: The breakdown of BC based eHealth studies

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
BC for hospital & drug management	Jamil et al.[159]	EHF	SC	✓	⋆	OfC	REST API, couch database, Hyperledger Fabric, Hyperledger Caliper	Cloud front-end Interface was developed to access the BC. Smart contract was designed for defining access policies to patient vital signs for the healthcare professionals.	Security and privacy concerns while transmitting vital signs to Blockchain have not been highlighted.
	Rathee et al.[232]	CPrB	CCM	⋆	✓	NM	NS2 simulator	Blockchain nodes are divided into two types: miner nodes and executing nodes. The executing nodes check the legitimacy of the Blocks	The configuration parameters and implementation procedures of BC on NS2 were not described
	Nguyen et al.[234]	EEB	SCM	⋆	✓	OfC	Ethereum Blockchain network on the Amazon Cloud	Cloud Blockchain was introduced to integrate EHRs to share data between healthcare professionals and patients.	How continuous health data can be handled on the Blockchain has not been covered.
BC for mHealth	Liang et al.[238]	PrPB	ECM	⋆	⋆	OfC/ OnC	Hyperledger Fabric	Blockchain was utilized to validate and preserve patient's data while sharing these with different stakeholders.	Real prototype was not implemented and privacy and security of IoT devices were ignored.
	Nguyen et al.[160]	ClB	SC	✓	⋆	OfC	Amazon web service, mobile android application.	Smart contracts based EHRs trustworthy control mechanism and data exchange protocol on Cloud Blockchain platform was developed.	Security and privacy analysis of the proposed system were missed
	Nguyen et al.[279]	NM	SC	✓	⋆	OfC	Not implemented yet	A mobile Blockchain was developed for clinical assessments and controlling.	Scalability and communication cost issues of the Blockchain have been not investigated

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, ✓= Yes, ⋆ = No

Table 2.10: The breakdown of BC based eHealth studies

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
BC for mHealth	Ni et al.[69]	NM	CCM	⋆	⋆	NM	Performance evaluation has not been carried out.	The authors developed an optimal decision-making process to keep BC mining cost effective.	Simulation has not been done to analyse its performance.
	Ichikawa et al.[239]	EHF	ECM	⋆	⋆	OnC	Hyperledger Fabric, JavaScript	The authors developed a mobile app to capture wearable sensor data to store those in private Hyperledger Blockchain	Performance has not been analysed in terms of throughput and energy consumption. The security issues between sensors, mobile App have not been addressed.
BC leveraged access control for eHealth	Tanwar et al.[40]	EHF	SCM	✓	⋆	OfC	Composer, Docker Container, Hyperledger Caliper, and the Wireshark	Access policies for healthcare entities were stored on the Blockchain. Algorithms defining access policy were designed.	Security issues such as malicious attacks and authentication were not addressed.
	Wang et al.[161]	CoB	SCM	⋆	⋆	OfC	Ethereum Blockchain	ABE(Attribute Based Encryption) was implemented using smart contract on Ethereum Blockchain.	The delay caused by ABE has not been addressed.
Storage of eHealth data	Liu et al.[247]	PuB	SC	⋆	✓	OfC	Prototype has not been developed.	EHRs were stored in the Cloud and index of EHRs were maintained on the Blockchain	Performance analysis has not been done.
BC based data sharing in eHealth	Xia et al.[75]	PrB	SC	✓	⋆	OfC	Simulation tools are not mentioned	An access management architecture that exploited smart contracts was designed to monitor access pattern of data users.	Extensive performance analysis has not been carried out.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned ✓= Yes, ⋆ = No

Table 2.11: The breakdown of BC based eHealth studies

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
BC based data sharing in eHealth	Dwivedi et al.[91]	CuB	NM	⋆	✓	OfC	Not implemented yet	The authors introduced overlay network for running Blockchain.	Simulation of the system was not done to analyze performances.
	Shen et al.[250]	NM	SCM	⋆	✓	OfC	WANem	Two separate networks named Blockchain and normal P2P network were designed. A session for packaging and removal of health data while sharing was also introduced.	Settings and configuration about Blockchain have not been described.
	Fan et al.[251]	NM	SCM	✓	✓	OfC/ OnC	breadcrumbs	The authors introduced Blockchain-based data sharing for hospital. The provincial hospital collects data from the community centres and participates in making Blocks.	Blockchain configurations are not discussed.
	Hang et al.[280]	EHF	SC	✓	✓	OfC /OnC	Hyperledger Fabric	The proposed scheme manages EMRs across different hospitals using Blockchain. Smart contract was designed to store data, logs and regulate access to data.	The implementation demonstrated the feasibility of the method.
BC based data sharing in eHealth	Zheng et al.[281]	NM	NM	⋆	✓	OfC	Not implemented yet	A conceptual model was outlined for sharing health data where the hash value of the data was stored on the Blockchain.	The real prototype of the framework is yet to be developed.
BC leveraged outsourcing in eHealth	Park et al.[169]	NM	NM	✓	⋆	OfC	Not implemented yet	A Blockchain based crowdsourcing platform was designed to provide data owner with incentives.	Performance of the proposed scheme has not been experimented.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned, ✓= Yes, ⋆ = No

Table 2.12: The breakdown of BC based eHealth studies

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
BC smart contract for eHealth	Daraghmi et al.[170]	EEB	CCM	✓	✓	OfC	Ethereum, Smart contract using Solidity language.	The authors developed a timed smart contract-based medical record access and permission management architecture. An incentive-based mining process was proposed to eliminate the need for digital currency.	The model was not designed for continuous patient monitoring data.
	Kazmi et al.[175]	EEB	SC	✓	⋆	OfC	Ethereum, Remix	Smart control manages and controls the enrolment of healthcare professionals and devices license in remote patient monitoring.	The security and privacy issues while retrieving data from wearable sensors were ignored.
	Malamas et al.[176]	NM	SC	✓	⋆	OfC	Not implemented yet	A medical forensic framework was proposed using Blockchain to save digital evidence and logs.	The prototype of the proposal was not implemented.
	Mytis et al.[180]	EEB	SC	⋆	⋆	OfC	Ethereum, MongoDB	The proposed system protects biomedical database queries using Blockchain technology	Different Blockchain related security attacks were not discussed.
Lightweight BC for eHealth	Ismail et al.[260]	CPuB	CCM	⋆	✓	OfC	NS3	A lightweight Blockchain was devised where only cluster head maintains Blockchain ledger.	The proposal cannot guarantee the tamper-proof of the data.
	Yang et al.[263]	CPuB	CCM	⋆	⋆	OfC	Not implemented yet	A novel context-aware consensus process called Proof of Familiarity was described to make a medical decision by gathering information from healthcare professionals and cured patients.	High-level performance analysis was done but the prototype of the proposal is yet to be implemented

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned, ✓= Yes, ⋆ = No

Table 2.13: The breakdown of BC based eHealth studies

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
BC leveraged searchable encryption for eHealth	Chen et al.[171]	EEB	ECM	✓	✓	OfC	Ethereum platform	A searchable encryption supported healthcare system using Blockchain was developed. The Blockchain contains the search index.	How BC based searchable encryption improved over the conventional was not demonstrated.
BC for eHealth architecture	Islam et al.[271]	NM	NM	⋆	✓	OfC	Not implemented yet	The authors developed a human activity recognition platform including Blockchain based Fog-Cloud computing.	The authors did not describe how Blockchain has been utilized in the proposed framework and no performance analysis has been conducted regarding Blockchain, fog and Cloud platform.
	Akkaoui et al.[162]	EEB	CCM	✓	✓	OfC	Go-Ethereum	A hybrid Edge Blockchain-based healthcare system has schemed where edge nodes certify the transactions and a separate global Blockchain stores metadata.	Using extra global Blockchain can increase the latency of processing Blocks.
	Calvaresi et al.[47]	EHF	ECM	⋆	⋆	NM	JADE-Java Agent, Hyperledger Fabric	Blockchain technologies (BCT) and MAS were combined to manage reputation for the Agents.	The performance of the Blockchain was not covered in the article.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned, ✓= Yes, ⋆ = No

2.2.2 BC Assisted Smart Cities/Home Management

The convergence of Internet of Things, Fog and Cloud computing has accelerated the advent of many sophisticated applications including eHealth, agriculture, supply chain, an automatic vehicle with the benefits of enhanced quality of services(QoS). This model can also increase resource utilization, and reduce operating costs.

With the advent of Fog, Cloud and IoT technology, a new business paradigm has evolved that enables customers to use cities/home's resources optimally to provide them with a wide range of services. Smart cities have a range of components including IoT systems, heterogeneous networks, large data storage and efficient information processing centres such as Fog, and Cloud server. Despite having such a vision of the smart cities, ensuring high quality and security for smart city services has appeared to be difficult. However, the Blockchain with attractive technological features, Cloud, and Fog computing can be a promising paradigm to opt smart cities/home services. Many recent studies indicate that Blockchain architectures can provide seamless connectivity between clients and industrial applications in smart cities. Recent studies pertaining to deploying Blockchain in smart cities has been summarized with respect to smart city and smart home service. Many of the current smart homes depend on third parties to provide different services to the resident, and in these systems, the resident has little control over his data. Cloud third parties store, process and manage the home data, which is often vulnerable to one single point of failure[282, 283].

Although the existing systems can provide smart home devices with fast connectivity and safe communication, those are centralised and have problems with scalability. Decentralized systems such as Blockchain and smart contracts have been regarded as a potential means of addressing these problems. A smart city is referred to an interconnected network consisting of computer servers, system administrations and other ubiquitous equipment such as IoT devices for capturing and processing all forms of data generated by city dwellers. Thanks to distinctive natures of IoT devices, the design of smart cities infrastructure remains challenges of ensuring anonymity, completeness, and to tackle bottleneck issues[284].

A collaborative framework for smart cities to ensure data integrity in the Cloud ecosystem was implemented in [181]. The architecture has two key entities: data owners and Cloud service providers (CSPs). To check the validity of data stored by different CPSs, they introduced a Blockchain-based auditing framework for users. In this context, Blockchain is used to develop a decentralised audit infrastructure which makes the overall system very stable and efficient without the need of third-party auditors. The weakness of the paper is that the implementation of smart contract and security assessment has not been done. The study in [181] discussed an authorization architecture and IoT delegation in Cloud-centric Blockchain project. The process is carried out using a smart contract that enables access control functions to ensure trust and auditing for network operations of users in IoT, and Cloud ecosystems.

In addition, a Blockchain was implemented in [285] to develop an IoT-based smart city infrastructure with three key components: smart node, P2P network and Cloud. Blockchain is unstable on IoT devices due to its resource constraints. The authors designed a lightweight Blockchain that requires low computational costs for the smart city infrastructure. All IoT devices' communications on a P2P Blockchain network are tagged as transactions and securely stored in Cloud storage. The architecture for smart cities retains five key cryptographic primitives, including authenticity and entry, confidentiality, and non-repudiation. However, the limitation of the work is that no access control has been designed for Cloud storage.

Meanwhile, Rahman et al.[172] has recommended a Blockchain smart contract-based shared

economy applications in the context of smart cities. The multimedia payload from the IoT ecosystem is uploaded and securely stored in IPFS distributed storage repositories as unchangeable headings. In particular, the system also provides a sustainable incentive mechanism that guarantees a secure cyber-physical sharing of IoT data. Smart contracts were implemented without the oversight of central authentication authority that ensures space-temporal services.

The smart home is a network of IoT devices with automated equipment, smart sensors, and detectors that capture environmental information from IoT devices to be stored on a Control server, particularly Cloud storage platform. While smart homes can provide residents with several advantages, there remain several challenges including malicious attacks and privacy issues to be resolved. Cloud computing powered by a Blockchain with distributed, secured and private properties[286] can provide a promising solution to these concerns.

Dorri et al. [287] suggested a smart home architecture that has three main levels: Cloud storage, overlay network, and smart home network. Intelligent tools were designed to handle transactions within the smart home and to preserve confidentiality, fairness and availability of IoT data. Data storage for the smart home network is managed via Blockchain-based Cloud service providers to provide high security for smart home operations. However, the shortcoming of the proposed scheme is that Blockchain for the system has not been implemented.

In [288], the integration of Cloud computing and Blockchain technologies provides a secure and efficient IoT smart home system. The system is composed of four general components: 1) smart home network, 2) Blockchain network, 3) Cloud infrastructure, and 4) application platform. Blockchain facilitated data traceability and Cloud server was exploited for distributed data storage. In addition, the system also offered recovery and trading facilities of the consumer data generated from the smart home network. Shared key policies were implemented on the Blockchain in order to guarantee smart home authorization and the availability of transactions between IoT devices and Blockchain miners.

In addition, Xue et al. [289] proposed a hypothetical access control system for home automation system, which includes a proprietary Blockchain to hold records of user transactions and store large-scale access data in off-chain storage, such as Cloud server. Singh et al. [290] proposed a smart home appliance management and controlling system utilizing Proof of Authority consensus mechanism of the Blockchain.

Ali et al. [291] implemented a Blockchain-based behavioural verification system for smart-IoT. The system demonstrated a degree of trust level for the external devices that want to join the smart home network. Blockchain was deployed in the IoT behaviour controller system to store, track, and identify IoT devices to safeguard IoT devices from malicious attacks. Sensor level filter has been utilized to prevent the malicious or faulty sensor from joining the network. Lee et al.[123] developed a Blockchain-based smart home architecture to solve the limitations of the existing centralised smart home network and combat future attacks against the smart Gateway. They used Ethereum Blockchain to make sure the smart home data was authenticated and confidential. The summary of some recent research in this field is illustrated in Table 2.14 and 2.15 respectively.

Table 2.14: The breakdown of BC based smart cities/home studies

Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
Ali et al.[291]	PrB	CCM	⋆	⋆	OfC/ OnC	Tensorflow and Keras libraries	A behaviour capturing, and verification procedures in Blockchain supported smart-IoT system were introduced. Blockchain was deployed in the IoT behaviour controller system to store, track, and identify IoT devices to safeguard IoT devices from malicious attacks.	Performance on the Blockchain has not been conducted.
Lee et al.[123]	PrB	CCM	⋆	✓	OfC	Mininet, Amazon EC2, Ethereum Bridge, Truffle development suite	A Blockchain-based smart home Gateway network architecture was proposed to overcome recent problems in current centralised security network architecture and combat future attacks on the smart homes Gateway.	The Gateway is vulnerable to a single point of failure and no approach was designed to tackle this problem.
Rahman et al.[172]	PrPB	CCM	⋆	✓	OfC	Amazon AWS platform, private Ethereum and Hyperledger Blockchain along with IPFS	The infrastructure leverage cognitive Fog nodes at the Edge to host and process offloaded geo-tagged multimedia payload and transactions. All result for AI processing is saved on the Blockchain and decentralised Cloud repositories to promote shared economy services.	The security and privacy issues of the offloaded tasks were not considered.
Singh et al.[290]	CoB	CCM	✓	✓	OfC	Cooja and Netsim, Amazon EC2	The Blockchain technology was used in a smart home network to manage system transactions and adopted green Cloud computing, which hosts a green broker to minimise the environmental impact of the model.	Blockchain configuration and settings for the simulators have not described in detail.
Yu et al.[181]	CuB	CCM	⋆	✓	OfC	Java Pairing-Based Cryptography Library (JPBC)	An automated blockchain platform called the blockchain data auditing (DAB) method, which gathers audit evidence was proposed. The DAB utilized a customized consensus algorithm based on the Practical Byzantine Fault Tolerance (pBFT) algorithm.	The authors did not describe how the optimized Blockchain was implemented.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned,
NA = Not applicable, ✓= Yes, ⋆ = No

Table 2.15: The breakdown of BC based smart cities/home studies

Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
Paul et al.[285]	CuB	CCM	⋆	✓	OfC	Ethereum, MySQL, DHT11 sensor	The authors adopted lightweight encryption for smart Blocks, such as symmetric key cryptography, which makes the smart Block more effective in terms of latency.	No access control has been designed for the Cloud storage.
Xue et al.[289]	PrB	CCM	✓	⋆	OfC	C language based on paired cryptographic library	A secure and auditable access control system for smart home using a private Blockchain was proposed	Full featured Blockchain was not designed.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned,
NA = Not applicable, ✓= Yes, ⋆ = No

2.2.3 BC Assisted IoT Vehicular Network

The recent development of sophisticated sensing, and computing devices, and information technology has resulted in significant growth in smart transportation services which have significant impacts on various aspects of our lives. Blockchain with Cloud, Fog and IoT can build a stable, reliable, and decentralised intelligent transport ecosystem. The integration of Cloud with virtually unlimited storage, Fog computing with processing capabilities and Blockchain with high-security feature revamps smart transport security and service quality. We reviewed smart transport applications into two main categories with respect to vehicle communication management and secure vehicle operation.

The incorporation of Cloud, Fog computing, and Blockchain can achieve efficient and secure connectivity in automated vehicular networks. Yin et al. [292] recommended a Blockchain-based multi-vehicle Cloud communication network to implement a structured framework. The private Cloud of vehicles from various manufacturers form a V2V(vehicle to vehicle) interconnected infrastructure using Blockchain decentralized system. Thus, the system facilitated various car services including asset management, sharing of ownership, co-operation and collaboration among private Cloud.

Liu et al. [293] implemented a layered architecture that comprises electric vehicles, Cloud and Edge network. The system created a pool of shared resources through facilitating collaboration among the heterogeneously-dispatched electric vehicles in order to provide seamless communications between heterogeneous entities. Blockchain was used to achieve robust security in sharing information and energy. In this context, a new Blockchain cryptocurrency for vehicular applications was proposed in which two kinds of the coin were introduced data coin and energy. The transactions generated in exchanging information and energy of the vehicular network are encrypted and added to a consortium Blockchain through a consensus mechanism.

Nadeem et al. [294] proposed a Blockchain Cloud-based vehicular distributed Ad-hoc (VANET) system to maintain the private lives of vehicle drivers with on-demand and low-cost access. The three interconnected components named 1) vehicle Cloud, 2) roadside Cloud(RSC) and 3) central Cloud form a Cloud hierarchical architecture to address the problems associated with VANET's storage, computation and broadband bandwidth constraints. The joint Cloud network securely links cars, service providers through a Blockchain regulated P2P network that can withstand cyberattacks and tackle bottlenecks issue in the car ecosystem.

Xie et al.[295] designed a Blockchain-based integrity management system for SDN-enabled 5G vehicular network. In this system, each vehicle shares a tag containing road information with other vehicles. The other vehicles nearby this sender offer it scores regarding the actuality of the shared information so that false or incorrect information cannot impact destination vehicles. The score providing vehicle determines the trust value based on their distance with the sender vehicle and put the value in the Blocks. Proof of Work and Proof of Stake consensus mechanisms were used to confirm the Blocks on the Blockchain. The simulation for the proposed scheme was carried out in OMNET++.

Michelin et al. [5] proposed SpeedyChain that decoupled Block headers from the Block's contents. The Blockchain for managing smart cities processes Blocks' headers on-chain. The authors set an expiration time while forming a Block to reduce its size and recommended the key update of the algorithms to minimize transactions' traceability. Further, they incorporated the level of access to control vehicles' permission. The experiment of the system was undertaken in an emulation environment using Common Open Research Emulator (CORE) to assess the performance of the SpeedyChain.

Meanwhile, Baza et al.[182] suggested Blockchain technology for providing autonomous vehicle (AV) with a firmware update on a regular basis. The manufacturers of AV inserted proofs into the on-board unit (OBU) of their AVs using ABE (attribute-based encryption) mechanism. The smart contracts were designed to hold policies about who has the right to download and use the firmware update. The authors proposed a Zero-Knowledge Proof in which each distributor exchanges an encrypted version of the firmware update with their AVs. The smart contract delivers the decryption key if the AVs can display the proofs obtained from the distributors.

The future transport system is going to accommodate driverless automatic vehicles that will carry freight and people. The human-driven gas station will be replaced with full autonomous electric charge station. In this scenario, transactions will be committed between machine to machine(M2M), and the present credit-based system is not adequate to facilitate transactions for such autonomous intelligent transport system. Pedrosa et al.[296] emphasized that the Blockchain technology can provide flexible and scalable facilities for M2M transactions targeting the use case of a driverless vehicle to be charged in electric stations. The shortcoming of the article is that the feasibility of the proposal was not studied.

Li et al.[297] proposed a Blockchain assisted vehicular Fog computing for carpooling services. Carpooling refers to the act of sharing a single vehicle with one or many passengers travelling in the same direction. The malicious users or drivers can falsely report their locations in such a system. To preserve passenger privacy and security, the authors applied conditional privacy, one-to-many proximity matching, destination matching, and data audibility in the carpooling scheme. The authors suggested Blockchain on RSU (roadside unit) that was deployed in the Fog layer. The Blockchain stores the hash of the data transactions generated from user's queries and the Cloud server stores those data. The queries, and report regarding car locations, route plan were passed to Cloud server via the RSU Blockchain so that malicious users cannot alter information. The experiment was conducted on private Blockchain.

Yao et al.[298] suggested a Blockchain assisted authentication approach for distributed vehicular Fog network. The authentication process was completed following four phases: 1) registration phase, 2) authentication phase, 3) consensus phase and 4) service delivery phase. In the registration phase, the on-board unit (OBU) of vehicles asks partial public key from the audit department (AD). The authentication phase involves the communications among OBU, vehicular Fog service (as known as RSU) and service manager (SM) for granting OBU access to resources. Next, the SM and WP (witness peer) run consensus protocol to insert the transactions of the authentication process into the Blockchain. The benefits of using Blockchain is that OBU does not require to initiate the authentication process next time when it moves to other data centre or Fog services.

Gao et al.[299] introduced a vehicular network which combined Blockchain, SDN and Fog computing. The vehicles equipped with OBU (on-board unit), RSU(roadside unit) and BS(base stations) perform the role of SDN data planes such as receiving packets, taking actions on these packets, updating counters and channel selection. On the other hand, the RSUH (roadside hub) was deployed in the Fog layer which acts as an SDN controller and decides the flow rules for the network. RSUH interconnects interzonal vehicular networks and runs Blockchain operations such as consensus mechanism. The Blockchain in the proposed scheme built a trust model by using information collated from peers to decide on messages to be sent from source vehicle to destination vehicle. The network parameter of the scheme was simulated on NS3 to analyse the performance in terms of packet delivery ratio and time. The Hyperledger Fabric was used to develop the Blockchain for the proposed 5G vehicular network.

In most recent research works of dynamic car parking allotment; the researchers have suggested VANETs where vehicles serve as hops to exchange information regarding the saturation status of

the parking lot. This approach encounters several challenges including sustainability and security because there is no incentive mechanism to exchange information with other vehicles and there is no consensus mechanism that can increase users' level of trust. To address this research gap, Hassija et al. [300] proposed a system based on DTL and DAG for allocating parking lots where DTL forms a protected peer-to-peer network with users, owners of parking lots, garages and free space. In the DAG network, a time-stamped consensus system was designed to process transactions related to requests for parking reservations in order to give users the best possible services in a cost-optimal manner. The authors also developed an adaptive pricing model for each parking request with respect to multiple parameters to provide the users with the best available slots in less time and expense.

State-of-the-art works in the Blockchain proposed a variation of Proof of Work to overcome the limitations of the generic Blockchain. PoS, PoB (Proof of Burn), and PoET follow the similar principle of PoW. Further, to apply PoS in a new distributed application is challenging because nodes in the network do not own any stake or cryptocurrency to burn in the initial stage. To address this issue, Hassija et al.[301] proposed a DAG-based energy trading platform for V2G(Vehicles to Grid) and G2V (Grid to Vehicles) where all transactions are stored in a tangle data structure. Further, a tip selection algorithm was devised to enable buyers and sellers to add new transactions in ledger without the need for miners. A game theory-based optimization algorithm was designed for both buyers and sellers to have the best deals in trading energy. The game theory guarantees a nash equilibrium between buyers and sellers, thereby preserving the price of energy sales.

The traffic jam prediction model assists users' vehicles to avoid congestion on the road. Such prediction model requires live traffic data, users' location and participants' private details including their name, and phone number that are sensitive. Google maps use crowdsourcing for data pertaining to live traffic congestion. However, not all users might be motivated to share their sensitive information about routes and locations with crowdsourcing without sufficient incentives. To realize this, Hassija et al. [302] suggested a traffic jam estimation system based on Blockchain where the Ethereum smart contract was designed to verify and store information from participants. The BC peer to peer network ensures safe sharing of confidential live traffic jam data from users. To estimate the probability of traffic jam at a specific location, an LSTM-based neural network was used. An incentive model that provides a user token if the user shares live traffic data with other users willingly. The user will use the token in the future to access the same services. Most studies in IoT vehicular that are reviewed above are summarized in Table 2.16 and 2.17 respectively.

Table 2.16: The breakdown of Blockchain based IoT vehicular studies

Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
Yin et al.[292]	NM	SM	★	✓	OfC	NA	Blockchain technology was integrated into the shared service model of the JointCloud network. Authors developed a series of information and value exchange networks that facilitate decentralised peer-to-peer communication between the various clouds.	The proposal is at conceptual and the Blockchain technology was not described.
Xie et al.[295]	CPrB	CCM	★	✓	OfC	OMNeT++, crypto++ library	SDN enabled 5G vehicular network was designed for trust management using Blockchain. A hybrid consensus mechanism based on PoS and PoW was also presented.	Setting parameters for Blockchain has not been discussed.
Nadeem et al.[294]	PrB	NM	★	✓	OfC	Not implemented yet	A Blockchain-based distributed Cloud architecture was proposed to safeguard the privacy of drivers.	The scheme was not implemented.
Baza et al.[182]	CoB	CCM	✓	✓	OfC	Python charm cryptographic library	Blockchain and smart contract-based firmware update scheme were proposed for AV's subsystem where a reward system was introduced to incentivize AVs to distribute the updates.	The authors did not describe how Blockchain was implemented.
Yao et al.[298]	CoB	CCM	★	✓	OfC	Java Runtime Environment	A Blockchain based lightweight anonymous authentication approach was proposed for distributed vehicular system.	The security protocol needs to be analyzed in enterprise BC.
Gao et al.[299]	PrB	SCM	★	✓	OfC	MATLAB, NS-3	The article highlighted the integration of Blockchain and SDN for the 5G enabled Fog vehicular network. A trust-based model is also provided to curb malicious attacks in the network	Integration of the three different technologies demonstrated a promising outcome.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned,
✓= Yes, ★ = No

Table 2.17: The breakdown of Blockchain based IoT vehicular studies

Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
Liu et al.[293]	CPrB	CCM	⋆	✓	OfC	Not implemented yet	A Proof of Work based on data contribution frequency and energy contribution amount was proposed in context-aware vehicular applications.	The context-aware Proof of Work has not been implemented.
Michelin et al.[5]	PrB	CCM	✓	✓	OfC	Common Open Research Emulator (CORE)	The Block header is decoupled from the Block contents and the Blockchain maintains Block header.	Performance analysis was done but how Blockchain was implemented in CORE was not described.
Pedrosa et al.[296]	EEB	SCM	⋆	✓	OfC	Not implemented yet	Refueling scenario for autonomous electric vehicles was described and an algorithm to ensure energy recharges was devised.	Prototype was not realized to analysis performance.
Li et al.[297]	PrB	CCM	✓	✓	OfC	Miracl cryptographic toolset	Privacy-preserving carpooling framework was devised using a Blockchain assisted vehicular Fog computing. The privacy of the users was guaranteed using on-to-many matching, destination matching and data auditability processes.	The privacy concerns in BC were addressed.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned,
✓= Yes, ⋆ = No

2.2.4 State-of-the-Art Works in BC Assisted Miscellaneous IoTs

This section contains the state-of-the-art works related to the Internet of Things beyond IoT healthcare, smart home/cities and vehicular network that were reviewed in the above sections.

2.2.4.1 Agent Managed BC in IoT

IoT devices cannot directly host Blockchain technology due to their limited processing and memory capacities. IoT devices produce vast numbers of transactions at a higher rate, but the current Blockchain cannot process those transactions at the same rate. Further, a large number of IoT transactions propagates in the Blockchain networks and causes higher energy consumption. To overcome these fundamental issues of integrating Blockchain into IoT ecosystem, Biswas et al. [303] proposed a scalable Blockchain framework that divides the Blockchain networks into two parts called local peer network and the global Blockchain network. The basic premise of this scheme and the scheme in [54] are similar. In both approaches, IoT systems are not be directly connected to the peer nodes of the Blockchain network. The proposed scheme in [54] described that the flow of transactions would be managed by using an intermediary agent between the devices and Blockchain peers as all IoT applications are usually affiliated with an organization. In [303], a local peer network is formed with devices from the IoT organizations to filter transactions and organize those in the Block. As a result, a lot of transactions remain within the local network that Blockchain would have to globally process. The approach in [54] differs from this scheme with respect to local agents' functionalities. Uddin et al.[54] proposed a local agent that dynamically determines storage repositories, governs the mining process of the Blockchain, maintains multiple Blockchain etc.

Calvaresi et al.[184] attempted to integrate Blockchain technology with the multi-agent system. Hyperledger Fabric was utilized as permissioned Blockchain, and the Agent was developed using JADE-Java Agent Development framework. The smart contracts manage the reputation of the Agent to measure its credibility. The users have been provided with a GUI to interact with agents in the system.

With Multi-Agent System (MAS), a software agent working on half of IoT devices is an efficient way to promote social interactions among intelligent devices. IoT devices need to associate them with a secure software agent when switching from one area to another [52, 53]. However, IoT devices do not generally have no accurate information available regarding the agents of a new environment. Further, IoT devices are often unknown, and unreferenced and the traditional approach of asking other trusted agents for information is usually impracticable. The work in [55] suggested a reputation model of the software agent in which the consumer's feedback for its services is summed up. Ethereum Blockchain was used to preserve and certify the reputation of all the agents in the distributed IoT networks.

2.2.4.2 BC for SDN Enabled IoT

Pourvahab et al.[304] proposed Blockchain leveraged forensic architecture in Software-Defined IoT network to collect evidence for the forensic experts to continue further analysis. The Blockchain was adopted into the control layer of the IoT SDN network to authenticate all the IoT devices for safe access. The SDN control layer runs a Neuro Multi-Fuzzy model to classify all kinds of packets into three categories based on six features. The features include the IP address of a source, IP address of a destination, length of flow, packet size, sequence number and type of operation. Finally, suspected evidence was stored in the Blockchain for future investigation.

El et al.[186] envisaged a Blockchain leveraged distributed Fog-Cloud architecture to provide IoT devices with secure, and on-demand access to the Cloud, and Fog. The Cloud-hosted Blockchain transactions and carried out Blockchain operations. The Fog layer incorporated Blockchain managed SDN-NFV technology to carry out computing resources for the IoT network. The SDN controller checked the availability of resources including computing, network, and storage pools at the request of NFV infrastructure servers. The SDN allocated the required resource and launched VNFs which is the basic block in NFVs architecture. The Gateway on the Fog layer provided the IoT devices with smart contract interfaces to place their requests and receive extra resources from the edge servers. The setbacks of the proposal are that the security concerns for the Edge servers were not addressed and the architecture was not implemented for analysing performances.

Talukder et al. [187] built a distributed database system using customized Blockchain technology to safeguard the system from malware attacks. Rathore et al.[188] also described a decentralized malicious attack detection and mitigation approach for IoT ecosystem with the aid of SDN, Edge, Fog, Cloud and Blockchain technology. The IoT devices are connected to SDN enabled Edge switch. The SDN-enabled switch records the information regarding dynamic traffic flow to assist the attack detection process executed at the Fog layer for finding suspicious traffic flows and blocking suspicious flows. The SDN controller at the Fog layer is connected to the SDN enabled switch. The SDN controller includes four components: 1) traffic flow analyser, 2) traffic flow classifier, 3) Blockchain-based attack detection, and 4) attack mitigation module. The first two components identify anomalous traffic and prepare an individual attack detection model for the Fog node. The third component contributes to the dynamic updating of the attack detection model using Blockchain technology and deep learning algorithm. The attack detection model helps the attack mitigation module to prevent attacks at the Edge layer. They implemented the framework on the Ethereum Blockchain and the Mininet emulator was used to analyse the performance of attack detection and mitigation approach. The authors described the design and workflow of the framework in detail and extensive performance analysis has been done. The significant contribution of the paper is that the Cloud agent collects local detection model using Blockchain smart contract and form a fusion attack detection model with higher accuracy. However, how Blockchain collaborated with SDN enabled switch and controller in the Fog layer could have been better described.

In Blockchain technology, all transactions generated in the P2P network is stored in a transactions Pool. The miners select a certain number of unconfirmed transactions from the Pools in a random fashion or based on fee in order to make a Block. The Block as well as transactions in the Block is verified and confirmed by the Blockchain nodes. However, this approach is not appropriate for transaction validation in time-sensitive service-oriented time tasks because the time-sensitive services might experience long latency. Hosen et al. [305] reckoned a context-aware selection process of transactions to be bundled into Blocks by the miners. The authors opted to add an extra field in the transactions to set priority for the selection process. The priority methodology figured out a transaction classification system based on a service's weight. The miner always picks the transactions with higher priority. An SDN-Gateway was introduced to bridge the lightweight IoT devices with Blockchain. The SDN-Gateway controls and collects transactions from the IoT devices. However, the challenge of this approach is to ensure that the weight on the service is honestly set because a client can falsely claim higher priority for its transactions. The author did not demonstrate how weight is measured for a transaction which has been left as future work. The performance of the proposed scheme was evaluated on a network emulator, called Common Open Research Emulator (CORE).

2.2.4.3 BC for Securing SDN IoT

Current IoT networking is confronted with many challenges including security, huge traffic, high availability, high reliability, high bandwidth, and limited energy. The recent emerging technologies such as distributed MEC (Multi-access Edge Computing), Software Defined Network (SDN), Network Virtualization Functions (NVF) and Blockchain are thought to address the existing challenges of the current IoT networking[306]. These technologies can either combinedly or individually meet the major IoT network requirements with high performance. Gao et al.[307] incorporated proxy-encryption (PRE) with Blockchain to improve IoT devices' credibility and authenticity in software-defined networking. Several smart contracts were designed to search and update records on the Blockchain. Mininet with OpenDaylight SDN controller was used to simulate the proposal. Hyperledger with Fabric SDK runs smart contracts to perform registration and enrolment of users.

In the Fog layer, Misra et al.[308] deployed a pBC (private Blockchain) where the Block content contains flow rules that are open to all SDN controllers. In the event of defective flow rules, the pBC allows for easy retraction to a previously running set of rules. As user's reliance on the Internet rises, traditional network designs with static characteristics will eventually fail to meet all requirements. SDN flexibility carries many security threats including unavailability of routing information to forwarding devices while forwarding devices fail to communicate with SDN controller due to the programmable interface being used illegally, and hidden vulnerabilities of a new complex system. To address the security issues, the Blockchain technologies are adopted at the SDN controller lever to record and scan network management information[309]. Distributed storage of Blockchain P2P enables to restore flow tables of a node during a network failure. To avoid unauthorized interference, Zhang et al.[309] suggested an information classification where the SDN controller's working process is managed by announcing the relationship of dependency between different type of information. However, how information classification was done has not been comprehensible.

Various network attacks are involved in SDN controllers, OpenFlow switches, and host interfaces. For instance, an infected controller can deliver misleading and deceptive instructions to OpenFlow switches by inserting and changing flow rules. Duy et al.[310] utilized Blockchain to record details about SDN events and actions in logs to perform digital forensics. To implement the SDN environment, the Floodlight controller as SDN control pane and Hyperledger Fabric Blockchain were integrated on Docker container.

Medhane et al.[311] proposed a security attack detection architecture for Software Defined Networking. The security architecture combined Blockchain, Edge, Cloud technologies for SDN enabled attack identification. The Cloud layer executes an algorithm to identify attack to reduce security attacks on the Edge layer. The SDN-enabled Gateway ensures dynamic monitoring of network traffic flows, which helps detect security attacks by determining dubious network traffic flows. The proposed security framework is implemented using Java programming and estimated performance in terms of network parameters including jitter, average energy consumption, packet delivery ratio, throughput, and delay.

Abou et al.[312] designed a Blockchain-enabled distributed DDoS attack mitigation framework. The framework utilized smart contracts on Ethereum Blockchain to pass attack information between SDN-based domains to promote attack collaboration securely and efficiently. The model is implemented on both private network (Ganache simulator) and the public network (Ropsten test network) of Ethereum to examine it in terms of versatility, efficiency, security, and cost-effectiveness. The smart contract written in Solidity language was deployed on the Ethereum Blockchain using the truffle platform. The smart contract was tested first on Ganache simulator

before installing on Ropsten's official Ethereum test network.

2.2.4.4 BC for Mobile IoT

With the exponential growth of mobile phones, the management of huge mobile data traffic requires a secure and fast network connection to boost QoS for consumers. However, network operators require heavy investment to constantly expand the capabilities of network infrastructure with the rapid rise of mobile devices. To cope with this issue, mobile data offloading to Fog or Cloud servers is a promising solution. The conventional algorithms for unloading mobile data do not have any mechanism to inspire or enable mobile devices or users to engage actively in the offloading process. Generic Blockchain was suggested to form a peer to peer network for facilitating mobile data offloading securely. However, the Blockchain with its conventional consensus mechanism lacks scalability and limit the performances of mobile data offloading process. Further, there involves a lot of microtransactions between the service providers and users in mobile data offloading. The users require to pay a certain amount to miner nodes for adding every microtransaction in the Blockchain which can demotivate them to participate in data offloading. To address this issue, Hassija et al. [313] suggested a lightweight framework based on Blockchain to enable mobile data offloading where the offloading is scheduled by a hashgraph consensus algorithm according to the minimum offloading time. The game-theoretical model was developed to negotiate and choose the best mobile devices in terms of computing power and processing time for data offloading. Their simulation results in the lowest cost of contact and suitable scheduling with other approaches to offloading.

Mobile devices can access the Edge servers to expand their computing capabilities. Edge computing has been seen as a promising solution for mobile Blockchain applications, which can bring several benefits. First, the robustness of the Blockchain network is enhanced, by adding more miners. Second, smartphone users can achieve a reward for executing a consensus mechanism of the Blockchain by utilizing Edge resources. To realize the economically benefited mobile system, the Edge providers need to set optimized pricing for the Edge computing services. For instance, Xiong et al. [314] suggested a pricing mechanism to buy Edge computing resources in mobile Blockchain network. The author in[314] introduced a mobile Blockchain network that allows mobile devices to invoke and access resources or computing services from the Edge network for running the mining process of the Blockchain. The mobile or IoT devices purchase computing resources from the Edge service providers using the two-stage Stackelberg game model. A prototype of the architecture was implemented to demonstrate important findings from the proposed pricing scheme.

Mobile and IoT devices are usually restricted to local computing resources. These devices require to offload computational tasks to the Cloud/Fog to perform Proof of Work. Jiao et al. [315] planned a Blockchain assisted auction mechanism for the resource-limited devices to utilize Cloud/Fog computing resources. The authors [315] suggested two bidding schemes: 1) the constant demand scheme in which each miner bids for a fixed quantity of resources, and 2) the multi-demand scheme in which the miners may apply their desired demands and bids. Further, they described an auction mechanism for the constant-demand bidding scheme which achieves the optimum total utility of the computation resources and the number of miners in the Blockchain network using an approximate algorithm. The authors designed their resource auction algorithm in Cloud/Fog ecosystem to address several questions such as to which miner the computing resources can be offered, what is the optimal number of miners because less number of miners can diminish the credibility of the Blockchain network and a large number of miners can cause network

latency, how fair pricing can be set for performing mining tasks. This model was implemented in Go-Ethereum platform to analyze performance.

Mobile devices can discharge data traffic to Fog layer to extend their network transmission bandwidths. The mobile devices can also discharge computing tasks to the Fog layer to release their workloads. Tang et al.[316] presented a Blockchain leveraged task offloading approach for the Fog-Vehicular environment. The Blockchain ledger saves the transactions related to the computational load of Fog servers. A vehicle chooses a Fog server based on computational load and distance. The proposed scheme was simulated on NS3. The work in [270] also envisaged a Blockchain-based task offloading approach for eHealth in Fog-Cloud ecosystem. However, the approach in [270] is different from the existing schemes in several ways. In [270], the Patient Centric Agent assists in outsourcing patient's tasks to a remote Fog Agent considering the sensitivity of the tasks. The computational parameters of the Fog Agent were divided as dynamic and static. Static execution parameter of a Fog Agent (such as CPU processing capability) is stored in the Blockchain while the dynamic execution parameters such as computational queue latency are asked from a group of Fog Agents. Storing transactions related to dynamic execution parameters increases power consumption and throughput of the task offloading approach. If more than one Fog Agents are the candidate for outsourcing, the Hungarian algorithm was used to optimize energy consumption and processing time. The remote Fog Agents can lie about their static and dynamic execution parameters. To prevent them from doing so, the Proof of Stake consensus protocol was modified to record reputation for every Fog Agents. The miners verify the static execution parameter by assigning sample tasks to a Fog Agent when a Fog Agent wants to join the Blockchain.

Nguyen et al.[317] has suggested a novel Blockchain mobile network in which smartphones load complex computing tasks onto the Edge node to facilitate computationally intensive mining. The article presented a privacy-preserving task-offloading network by considering the complexities of the Blockchain transaction states and channel states between the miners and the Edger server. They proposed an optimal DRL-based algorithm for all miners by using a deep Q Network to achieve complete confidentiality and reduce the cost of latency and resources.

2.2.4.5 BC for Wireless Sensor Networks

In wireless sensor networks, nodes remain unattended for an extended period and often fail to properly operate due to natural disastrous and malicious attacks. To recover the failed nodes, Noshad et al.[318] proposed a Blockchain-based node failure detection and recovery approach in a wireless sensor network. In this process, the hierarchical structure of the nodes was considered where a cluster head (CH) maintains the Blockchain ledgers. If a cluster head goes in an active state, a centralized entity requests the session history of the failed node from other CHs which degree of nodes is higher. The authors introduced a smart recovery contract (SC) to record the state of every CH.

Yazdinejad et al. [122] schemed a Blockchain-based decentralized authentication process for the Internet of Underwater Things. The cluster head chosen from IoUT devices forms a P2P network for running the Blockchain. If a node in a cluster is approved, the node can authorize other nodes and be trusted in other clusters. The node does not need to perform the authentication process again while communicating with other devices in another cluster. The Blockchain ledger contains a unique device and other information regarding IoUT nodes.

Uddin et al. [319] contributed to exploring a smart Agent's feasibility in tracking underwater IoT and IoT smart home or cities using a custom Blockchain. In Blockchain leveraged underwater IoT monitoring framework, they designed a secure light hierarchical routing protocol for

the underwater sensors deployed at different depths and a lightweight consensus mechanism of the Blockchain for processing underwater IoT data. Java programming was used to implement the system. The architecture consists of three layers: Underwater IoT layer, the Edge layer and Cloud layer. The smart Agent residing in the Edge layer receives data from the surface nodes of the IoUT layer and selects a group of suitable Miners from Cloud Blockchain network using the TOPSIS method to process IoUT data. To analyse the efficiency of the proposed consensus protocol in detecting an anomaly, the authors used publicly available datasets called KDD Cup 1999 Data[130]. In addition, the performance of the Blockchain-based routing protocol is evaluated in terms of different metrics such as block time generation, energy consumption, remaining energy and reliability.

Pop et al. [189] explored the Blockchain technology in a smart grid to manage demand response of energy. The smart contracts executing on the Blockchain defined the expected levels of energy demand, validated demand response agreements, and a balance between energy demand and production. The Ethereum Blockchain was used to implement a prototype of the smart grid based on UK building datasets. Cech et al.[320] investigated the full functionalities of Blockchain on the Fog network to share data emitted from IoT sensors. The authors used virtualized features of the Blockchain using docker container orchestration and management system with its Swarm mode and MultiChain framework. The prototype of the proposed scheme was designed on a Raspberry Pi SBC testbed to show the viability of the data sharing with higher security and integrity.

Zhu et al.[321] investigated Blockchain in the Fog layer to set up a social network which managed two main services: Identity management and relationship management services. They further outlined access policies based on the relationship of the users. The authors described the identity registration, update, and revocation process on the Blockchain-enabled Fog network. The prototype of the system was implemented using SELinux and a Raspberry PI as a Fog node.

In IoT Fog computing ecosystem, the PoW consensus mechanism is not appropriate due to its high-power consumption and time. Kumar et al. [221] optimized PoW for the IoT-Fog network using statistical method. They used polynomial matrix factorization to reduce the number of iterations to find the solutions for PoW. The proposed scheme was implemented to demonstrate the power consumption and processing time.

Biswas et al. [32] presented a lightweight consensus mechanism called the Proof of Block Trade (PoBT) for validating diverse kinds of trades. The authors incorporated this consensus mechanism into the architecture of the Hyperledger Fabric to build a scalable local trading network. Samuel et al.[322] presented a Blockchain-based data-sharing model for the smart grid which also included a Proof of Authority(PoA) consensus mechanism using page rank algorithm to minimize gas consumption and computational cost. In addition, Huang et al. [8] proposed a self-adaptive PoW algorithm to reduce power consumption for the power restricted IoT devices. The authors suggested determining the difficulty level of PoW consensus mechanism considering the nodes' behaviour in which difficulty level was reduced for the honest node and was increased for malicious nodes. An access control scheme which uses a robust data authority management approach based on symmetric cryptography in a transparent Blockchain network was also explored in the work.

2.2.4.6 BC for IoT Supply Chain

Hassija et al. [323] presented a thorough analysis of security and privacy problems relevant to various supply chain management areas. Three technologies-Blockchain, machine learning, and PUFs have been identified as a way of resolving security threats and other problems prevalent in traditional supply chains. For prospective researchers, the possible scope of study and recommen-

dations for the supply chain have been addressed.

Malik et al. [183] developed a TrustChain which is a Hyperledger Fabric Blockchain-based supply chain system. The research included reputation management at each level of the supply chain from the product to the consumption, including the role of supply chain entities. The authors leveraged the smart contract to automate the assessment of reputation based on the quality of the food product being traded, the trustworthy of the supply chain participants and penalized the participants that withdraw their roles and falsely circulate high ratings. Malik et al. [324] also devised a tiered architecture to maintain provenance in supply chain system and facilitate a forum for collaborating between supply chain entities and administrative bodies. The framework included an Access Control List (ACL) for transactions' reading and writing access and managed a set of parallel Blockchains instead of one large Blockchain. A transaction vocabulary was introduced to link the final product with multiple raw ingredients.

Figorilli et al. in [325] presented the application of Blockchain technology to manage the wood supply chain. The authors adopted RFID technology on the wood to receive information from various stages of wood processing, including from standing trees to the final products, going through cutting, felling, harvesting and sawmilling process. They simulated the wood supply chain in the Calabria of Southern Italy.

Hang et al.[280] investigated Blockchain in the agricultural sector. The Blockchain algorithms such as consensus algorithm consume higher power consumption and cause high latency in confirming Blocks. The complete replacement of the legacy system using Blockchain requires to invest enormous resources. That's why the authors suggested a hybrid architecture for tracking fish from the production to consumption by combining the legacy system and Blockchain technology.

2.2.4.7 BC Based Authentication for IoT

Manzoor et al. [173] designed a hybrid architecture for IoT data sharing by integrating Blockchain, smart contracts and Cloud. The storage problem on the Blockchain was solved by using Cloud storage. The proxy-encryption scheme was used as the security mechanism to enable only the owner and individuals listed in the smart contracts to access the data. A testbed was implemented to check the feasibility of a platform with respects to scalability and performance metrics.

Martinez et al. [326] enhanced the authentication scheme proposed by Zhou et al.[327] to prove the legitimacy of a member in the network. The authors [326] added a new sub-phase called link attempt evidence to identify the authenticity of the participant.

Xu et al.[177] recommended Blockchain in IoT ecosystem for authentication and verifying reliable services from untrusted Edge entities. The Cloud service providers stores services or program code on off-chain and business security-related transactions on-chain. The Edge entities which are at one hop away from the lightweight client cache the services or program code for IoT devices. The lightweight client requests a service from the associated Edge entity and triggers a smart contract to verify the authenticity of the services on the Blockchain. They analysed the performance of the proposed work using Ethereum Blockchain.

Ma et al.[328] advanced a Blockchain-based distributed key management architecture which includes Fog, and Cloud computing for guaranteeing hierarchical IoT access controls. The Fog network containing a security access manager (SAM) is divided into different zones. They break the Blockchain into various side Blockchain to save storage for IoT applications. Each SAM manages a side Blockchain for its domain. The Cloud collects all side Blocks from every SAM and hosts multi-Blockchains to facilitate cross-domain interactions. The proposed scheme was implemented in OMNet++ to analysis security strength and transaction processing time.

Almadhoun et al. [178] sought Edge servers to perform authentication using a smart contract on the Blockchain on behalf of IoT devices. The Fog nodes have an interface with Ethereum Blockchain's smart contract to relieve the burden IoT devices from running an authentication process. User can access IoT devices via Blockchain-enabled Fog servers connected with Ethereum smart contracts.

Nguyen et al. in[179] utilized smart contracts to ensure that the authorized users can access data without the requirement of third parties. The authors also projected a firmware update scheme for the IoT devices by leveraging Blockchain to avoid fraudulent and data tampering.

In addition, Bao et al. [329] presented an IoTChain which is consisted of three layers: 1) authentication layer, 2) Blockchain layer and 3) application layer. The architecture provides several services such as identity authentication, access control, the integrity of storage without incurring high overheads and delays. They claimed that the architecture offers a lightweight feature, and fault tolerance to DoS attacks.

2.2.4.8 BC for IoT Trust Management

Kochovski et al.[190] developed a trust management system in Edge-Cloud orchestrated network using Blockchain. The system consists of four layers: 1) application layers, 2) Blockchain layers, 3) decision-making layers, and 4) Edge to Cloud orchestration layer. The devices from Edge to Cloud orchestrations need to register to the Ethereum Blockchain. The smart contract on the Blockchain manages the trust for each device based on the author's defined attributes. The decision-making layers select an Edge service or Cloud services using Markov decision process considering QoS. The strength of the proposal is to develop smart contract to measure the trust of IoT, Edge and Cloud devices based on user's subscription and several attributes. However, the performance analysis of the Markov decision process for selecting Edge-Cloud providers is partially completed. The Blockchain has been used for only maintaining the trust of Edge-Cloud but other security requirements such as data integrity, confidentiality and availability have not been addressed.

Debe et al.[185] designed a decentralised trust model to ensure the credibility of Fog nodes while IoT devices request computing services from the Fog nodes. The reputation scores for a Fog node is computed based on the client's opinion about their previous interactions with public Fog nodes. The nodes that frequently provide ratings have more influence on the reputation of Fog nodes. A client is penalized if it provides false ratings to a Fog node. The Ethereum Blockchain stored the reputation for the Fog nodes. The proposed scheme was tested by developing different smart contracts on the Ethereum. The smart contracts for registration, computing reputation scores and credibility of Fog nodes are tested on Remix IDE using solidity language. Remix IDE is an online tool to develop, debug and test code on a virtual Ethereum Blockchain. The front end of the Blockchain layer has been built on the Truffle Suite. The work in [270] also incorporated trust model using a modified page ranking algorithm. Further, The research in [270] included reputation in a modified Proof of Stake consensus algorithm in Fog-Cloud network.

2.2.4.9 BC for IoT Payment Management

Customers need to pay Cloud service providers for outsourcing their tasks. In the traditional system, customers subscribe to Cloud services using different banks. In such a system, both clients and servers require to trust third parties for guaranteeing services and payment which causes bottleneck and distrust problems. Zhang et al.[281] proposed a Blockchain-based payment system (BCPay)

for outsourcing resources from Cloud and Fog. The BCPay system includes clients, Cloud server and a Blockchain. The BCPay's operations are performed in five phases: 1) set up phase 2) service implementation phase 3) service checking phase 4) service payment phase 5) service demand phase. All these phases involve Blockchain for completing the payment process without the need for third-party trust. The strength of the proposal is to analysis the performance of the security protocols of the proposed payment system. However, Blockchain-based implementation has not been done and which kind of Blockchain used was not highlighted. Further, the BCPay was not investigated for Fog computing. Debe et al. [174] also designed Ethereum Blockchain-based monetization and automated payment for public Fog nodes. The IoT devices can pay to their connected public Fog nodes for the services via an automated dispute-free payment system controlled by using a smart contract. They tested the proposed scheme using the similar settings of their earlier research in [185].

Meanwhile, Pan et al.[36] developed an EdgeChain, a Blockchain and smart contracts leveraged framework for the Edge-IoT network. The system utilized an internal currency for purchasing IoT services from Edge and Cloud. IoT devices used credit-based coins to purchase Edge servers where smart contracts applied regulation enforcement mechanism to control the actions of the IoT devices. They implemented a prototype to test and evaluate the EdgeChain. Furthermore, Seitz et al.[330] described a case study of IoT marketplaces which included Blockchain and Fog computing to make IoT services available to clients. The customers can check an App on the Blockchain if it is available there. If so, the customer places an order on Fog node using smartphone interfaces and the Fog node brings the app from the storehouse.

The succinct analysis of miscellaneous IoT and Blockchain related studies are presented in Table2.18, 2.19, 2.20, 2.21, 2.22 and 2.23 respectively.

Table 2.18: The breakdown of BC assisted IoT works

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
Agent managed BC in IoT	Biswas et al.[303]	EHF	SCM	⋆	✓	OfC/ OnC	Hyperledger Fabric, kafka-Zookeeper, Configtxgen	The authors introduced local peer network to connect IoT network with global Blockchain which can limit the number of transactions entering into global Blockchain.	Local peer is vulnerable to many cyberattacks including Ransomware, and DoS attacks
BC for SDN enabled IoT	Pourvahab et al.[304]	CPrB	CCM	⋆	✓	OfC/ OnC	NS3, Python, C++, OpenFlow switch	A forensics architecture that adopted Blockchain network on SDN controllers for implementing chain of custody.	The paper did not describe in details how different tools are integrated to implement the proposal.
	El et al.[186]	PrB	NM	⋆	⋆	OfC	Not yet implemented yet	An architecture combining Blockchain, Edge computing and IoT was described.	The conceptual model was proposed without performance analysis
	Rathore et al.[188]	EEB	SCM	⋆	✓	OfC	Mininet, Amazon EC2, Ethereum, Truffle development suite	SDN enabled Fog computing, Cloud and Blockchain technology were combined to detect attacks in the IoT network.	The authors did not describe how Cloud and Blockchain technology was integrated into Mininet tools.
	Hosen et al.[305]	CuB	CCM	⋆	⋆	NM	Common Open Research Emulator (CORE)	A context-aware transaction validation mechanism for the Blockchain's miners was proposed where the miners select transactions from the Pool with the priority of service.	The author did not demonstrate how weight is measured for a transaction which has been left as future work.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned, NA = Not applicable, ✓= Yes, ⋆ = No

Table 2.19: The breakdown of BC assisted IoT works

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
BC for mobile IoT	Xiong et al.[314]	PrB	SCM	✓	⋆	NM	Intel Xeon CPU E5-1630 as Edge node	A prototype of mobile Blockchain network was simulated where the mobile devices or users can access and utilizes computing resources from the Edge service providers using two-stage Stackelberg game theory to run PoW consensus mechanism.	PoW consensus mechanism demands high power consumption and causes a delay in the mining process. The author could investigate other consensus protocol such as PoS.
	Jiao et al.[315]	PrB/ EEB	SCM	✓	✓	OfC	Docker platform, Go-Ethereum	The authors proposed an auction-based market model to trade between the Cloud/Fog computing services and Blockchain miners regarding purchasing resources.	Although optimization of mining process improves network performances including bandwidth, power and storage, it makes the mining process less decentralized and vulnerable to cyberattacks.
	Tang et al.[316]	PuB	SCM	⋆	✓	OfC	NS3	The authors incorporated Blockchain in the Fog network to facilitate secure task offloading	The authors did not focus on privacy of offloading tasks.
	Nguyen et al.[317]	EEB	SCM	⋆	✓	OfC	Ethereum, Lambda Edge, Amazon cloud, Biokin sensors	The authors proposed a task offloading for Blockchain assisted mobile Edge computing network using Markov decision, reinforcement learning (RL) and deep RL Q-network where mobile users act as miners and outsource tasks to Edge server.	The authors performed an extensive experiment and evaluated different performances that showed the approach's feasibility.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned, ✓= Yes, ⋆ = No

Table 2.20: The breakdown of BC assisted IoT works

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
BC for wireless sensor networks	Noshad et al.[318]	PuB	SC	⋆	✓	OnC	Remix IDE, MetaMask, Ganache, Rinkeby test network and MATLAB R2018a	The authors suggested a Blockchain-based node recovery method for WSN where failed node is recovered based on the node degree.	Security strength of the approach was not evaluated.
	Yazdinejad et al.[122]	PrB	SCM	⋆	⋆	OnC	NS2	A decentralized authentication using Blockchain for underwater sensor networks was proposed	The authentication protocol was not described well and the role of the Blockchain in this process has not been clear.
	Uddin et al.[319]	PrB	CCM	✓	✓	OnC	iFogSim, Java Programming	The authors designed an Blockchain based multilevel architecture for Internet of Underwater Things	Security analysis has not been carried out in the simulated environment. Instead, high level security analysis has been done.
	Pop et al.[189]	EEB	SCM /SM	⋆	✓	OfC /OnC	Ethereum platform	The author exploited Blockchain to build a smart gird for handling energy demand response.	The customer's privacy was not addressed.
	Cech et al.[320]	PuB	CCM	✓	✓	OfC	Raspberry Pi SBCs	The authors built a Fog computing system called HCL-BaFog using Blockchain to collect and exchange sensor data safely.	Full featured Blockchain might not be supported by all kinds of low-profile Edge nodes.
	Zhu et al.[321]	PuB	SCM	✓	⋆	OfC	SELinux, Raspberry PI	Fog computing and Blockchain to build a trustless social network system was investigated.	User's privacy has been addressed using access control.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned, NA = Not applicable, ✓= Yes, ⋆ = No

Table 2.21: The breakdown of BC assisted IoT works

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
Optimization of BC consensus method	Kumar et al.[221]	CPuB	CCM	⋆	✓	NM	NM	The authors devised a modified PoW for Cloud and Edge computing using expectation maximization algorithm and polynomial matrix factorization.	Tools to implement for the Blockchain have not been mentioned.
	Biswas et al.[32]	PuB/ EHF	CCM	⋆	✓	NM	Hyperledger Fabric	The authors proposed a lightweight proof of block and trade (PoBT) consensus to optimize Proof of Work	The performance evaluation shows that without sacrificing security, the proposed consensus mechanism can reduce power consumption.
	Huang et al.[8]	CuB	CCM	⋆	✓	OfC	RESTful HTTP, RPC, IOTA Python API	A credit-based Proof of Work was proposed where the difficulty level is reduced for honest nodes and increased for malicious nodes.	Variations in the degree of complexity will increase the risk of Blocks in the ledger being manipulated.
BC for IoT supply chain	Malik et al.[324]	CoB	SCM	⋆	✓	OfC	Hyperledger Composer, Caliper	The authors built a consortium Blockchain trust management system for the supply chain where trust and reputation scores for the participants are determined based on their interactions	The sharding technique improved the performance of the system.
	Figorilli et al.[325]	CoB	SCM	⋆	⋆	OfC /OnC	Azure Blockchain Workbench, MySQL server, REST API, JSON	The authors implemented an electronic traceability system using Blockchain where RFID sensors and open source technology were used for info tracing.	Appropriate security and privacy methods are needed at every stage of the traceability system.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned, NA = Not applicable, ✓= Yes, ⋆ = No

Table 2.22: The breakdown of BC assisted IoT works

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
BC for IoT supply chain	Hang et al.[280]	EHF	SCM	✓	✓	OfC	Couch DB, Hyperledger Fabric, Docker engine, REST API, JSON	The combination of Blockchain and conventional system was investigated to store agriculture data from the fish farm in tampered proof way.	The conventional portion of the system is still vulnerable to cyberattacks.
BC based authentication for IoT	Manzoor et al.[173]	EEB	SCM	⋆	⋆	OfC	Ethereum	A Blockchain based proxy re-encryption scheme were presented. Experiment was done on Ethereum Blockchain	Details about settings and parameters are missing in the experiment.
	Ma et al.[328]	PrB	CCM	✓	✓	OnC	OMNeT++, ECIES, curve secp160r1	The authors proposed a novel multi Blockchain based Fog-Cloud architecture for managing security key.	Although multi Blockchain improves performances, a chain might be manipulated and recreated by malicious attackers
	Almadhoun et al.[178]	EEB	ESM	✓	⋆	OnC	Remix IDE, Solidity language, Ethereum	An authentication mechanism for Blockchain enabled Fog network where Edge servers facilitated interface to access IoT devices via smart contracts on the Blockchain	implementation of prototype is left as future work.
BC for IoT trust management	Kochovski et al.[190]	EEB	SCM	⋆	✓	OfC	Ethereum	A trust management architecture for Fog-Cloud was implemented using Smart Contracts.	The Blockchain has been used for only maintaining trust of Edge-Cloud but other security requirements such as data integrity, confidentiality and availability have not been addressed.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned, NA = Not applicable, ✓= Yes, ⋆ = No

Table 2.23: The breakdown of BC assisted IoT works

Category	Authors	1	2	3	4	5	Tools/Simulator	Contributions/Outcome	Weakness/Remarks
BC for IoT trust management	Debe et al.[185]	EEB	SC	⋆	✓	OnC /OfC	Ethereum, Remix	The authors proposed a Blockchain reputation model for public Fog nodes-based user's opinion about their past interactions with the public Fog nodes.	The performance of the system was not analyzed with respect to power consumption, throughput and other parameters.
BC for IoT payment management	Zhang et al.[281]	EEB	SM	✓	✓	OnC	Ethereum	The author presented a Blockchain based payment system called BCPay with architecture, specifications and adversary model	Blockchain based implementation has not been done and which kind of Blockchain used was not highlighted. Further, the BCPay was not investigated for Fog computing.
	Debe et al.[174]	EEB	SC	⋆	✓	OnC	Ethereum, Solidity language, Remix web tools	The authors proposed a Blockchain-based monetization and payment system for the public Fog nodes for the services they provide.	The performance of the system has not been analyzed with respect to power consumption and throughput.
	Pan et al.[36]	EEB	SCM	⋆	⋆	OnC	OpenStack, Go-Ethereum, Truffle	The authors incorporated a permissioned Blockchain to link Edge Cloud resources with IoT devices using internal coin currency.	Trust module is yet to be included in the system to make it sustainable.
	Seitz et al.[330]	EHF	SCM	⋆	⋆	OfC	NA	The authors recommended an IoT Bazaar to trade Edge apps using Blockchain to enable the monitoring of app installations on Edge devices.	Performance analysis has not been conducted for the proposal.

1 = Blockchain type, 2 = Consensus protocol, 3 = Access control, 4 = Scalable, 5 = Storage, NM = Not mentioned, NA = Not applicable, ✓= Yes, ⋆ = No

2.3 Conclusion

We reviewed research from several domains including IoT eHealth, smart home, smart vehicular applications which incorporated Edge, Fog, Cloud computing and Blockchain technology to address security and privacy challenges. Nonetheless, a variety of technological and security issues in IoT remain unaddressed. In this chapter, several challenges in undertaking Blockchian technology in IoT domain are identified and discussed how those are addressed. The existing Blockchain and IoT articles were scrutinized with respect to diverse attributes for demonstrating their strength and limitations. Further, the chapter includes a broad description of Blockchain components and several standard consensus mechanisms.

Chapter 3

A PCA Managed End to End Secure Customized Blockchain Based IoT eHealth Framework

Blockchain technologies bring several attractive benefits to eHealth such as decentralized, tamper-proof management of health data without the oversight of a third party. However, the integration of Blockchain with body area sensors raises many issues:

- designing scalable eHealth framework
- unbalanced processing rate between the sensors' data streaming and Blockchain miners
- patient's privacy concerns due to the transparent nature of Blockchain
- Blockchain's incapacity to accommodate a large volume of data
- sensor's limited processing and storage capacities to execute Blockchain security and mining algorithms.

To meet these challenges, in this chapter, we introduced a Patient-Centric Agent that is a software agent running on dedicated hardware to connect Body Area Sensor Networks with a Blockchain peer to peer network. The Patient-Centric Agent incorporates three significant modules: medical data management, Blockchain mining management and security management. Each of these modules includes different sub-services based on the patient's needs and preferences. Previous studies that attempted to integrate IoT with Blockchain technology suggested a smart Gateway to coordinate the IoT ecosystem on behalf of users. However, a Gateway service is limited in managing security for IoT devices. In contrast, our Patient-Centric Agent is envisaged to manage health data, multiple Blockchains and patient privacy and security. The Patient-Centric Agent organizes transactions into Blocks which increases throughput in the Blockchain and reduces the number of transactions propagating in the Blockchain network. We advanced an End to End secure eHealth framework with the inclusion of the Patient-Centric Agent that bridges the gap between Body Area Sensor Networks and a customized Blockchain. Our main contributions include the following design elements:

1. We designed a two-tier End to End secure eHealth architecture (depicted in Figure 3.3) where the upper layer deals with administrative activities, and the lower layer handles data streaming and storage. The upper layer contains the HCU(Healthcare Control Unit) that produces public parameters and cryptographic keys for healthcare professionals. The HCU handles legitimate complaints against users and healthcare professionals. The HCU reveals a target user's identity and bans him or her. The HCU's function does not conflict with the Blockchain as it only serves to initialize parameters and disclose healthcare professionals' identity.

2. In the lower layer, we developed a lightweight authentication protocol for establishing secure communication between different components of the eHealth system including the Patient-Centric Agent, Blockchain and body area sensors. The key features of the protocol include dynamic key generation and the integration of proximity authentication with a lightweight HMAC(Hash MAC). The advantage of using dynamic key is that in future, an attacker will not be able to hack communication channels repeatedly using the same key. The justification for using Hash-based proximity authentication is that a malicious device may not hack a patient's device due to its geographical locations. Furthermore, Hash-based authentication consumes lower power, making it ideal for IoT devices.

   The Patient Centric Agent also incorporates a Hash oriented role-based access control mechanism. The PCA makes separate lightweight secret symmetric keys for producing ciphertext of data on every Block. To provide healthcare providers with access to patient's data, the PCA creates an access grant transaction which includes an access control code and access grant expiration time. As a result, a healthcare provider cannot gain access to a patient's data for an indefinite time once the healthcare provider obtains patient's secret key. Furthermore, additional security and data anonymity is ensured because patient's data on each Block is encrypted using a distinct secret key. The flow diagram illustrated in Figure 3.1 presents the communication protocol described above.

3. We customized a private Blockchain with the aid of the Patient-Centric Agent to facilitate health data on the Blockchain and optimize Blockchain in terms of power consumption. The Blockchain is customized in the following ways.

   (a) The Patient-Centric Agent governs the mining process by selecting a group of miners to compete to add a Block whereas conventional consensus algorithms allow all miners to compete. A key feature of the selection algorithm includes a hybrid consensus algorithm that integrates Proof of Work and Proof of Stake approaches with a measure of the miner's reputation in the form of trust. With respect to energy consumption and processing time, this approach can substantially increase efficiency over the traditional consensus mechanism. However, because a single Miner is responsible for computing the target hash code of a Block, the consensus mechanism is vulnerable to cyberattacks such as Sybil attack, single point of failure, and selfish mining attacks. To avoid these attacks, a set of potential Miners first are nominated using Proof of Stake and Proof of Capacity. Second, the Miners' rating is estimated using Miners' trust value. Finally, a secretary algorithm selects a Miner based on Miners' rating to perform Proof of Work. The Proof of Stake entails that every Miner must deposit certain amount to participate in validating Block. So, a Miner found to be malicious will lose its deposit which prevents them from engaging in selfish mining.

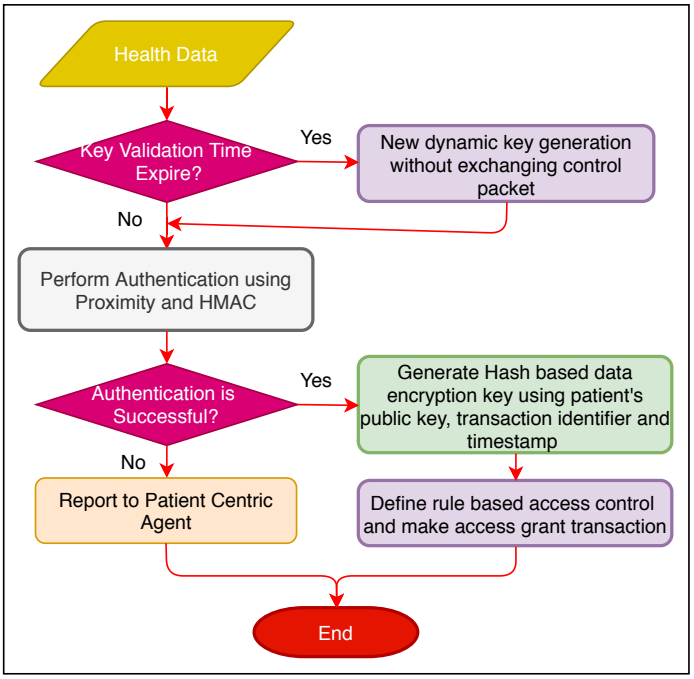


Figure 3.1: The communication protocol of the proposed IoT healthcare system

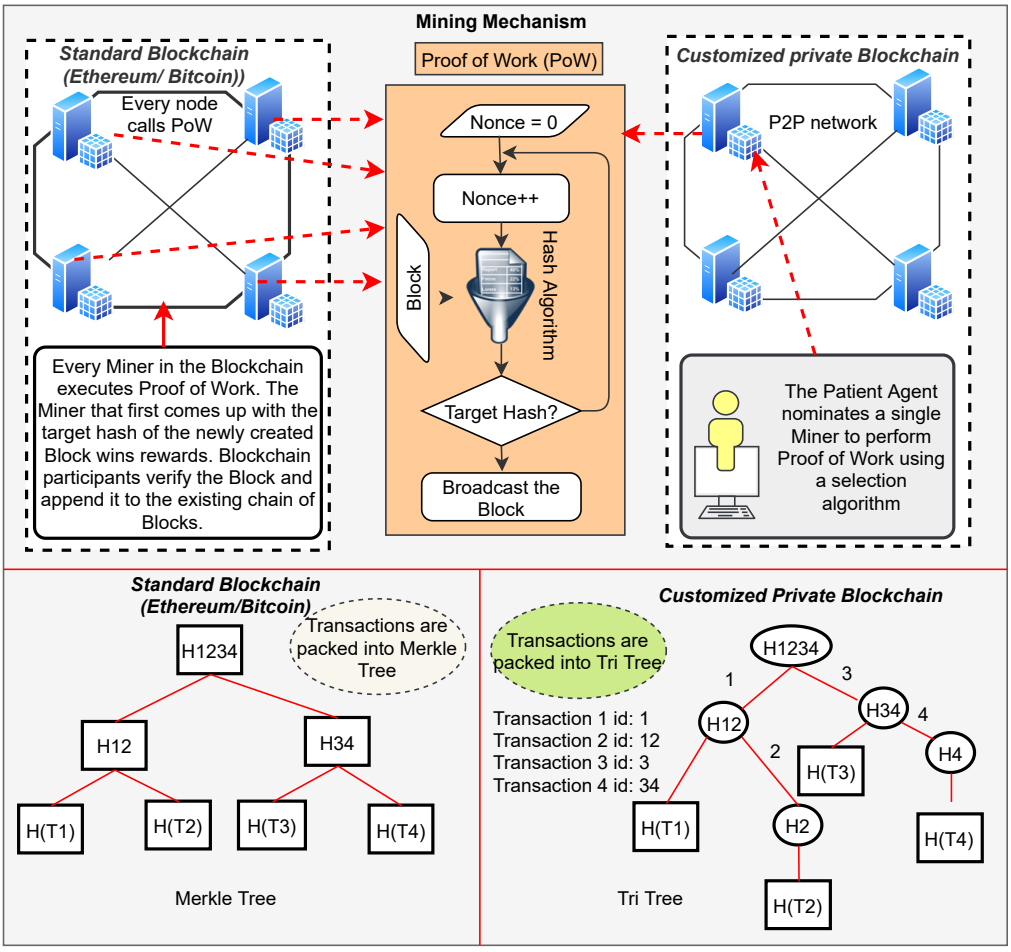


Figure 3.2: The Bitcoin and customized private Blockchain

The trust value is estimated based on a range of Miner's properties including their availability, mining rate, processing capacities, previous records regarding their honesty etc. This trust value can filter malicious Miners. The most fitted Miner is discovered based on rating using secretary hiring algorithm which provides some degree of randomness. As a result, the same Miner has very little chance to be repeatedly elected. Given the adversarial chosen candidates, the secretary hiring algorithm[331] is a technique to maximise the likelihood of finding the best miner candidate in a randomly selected order.

(b) The Blockchain is further modified using a Tri tree instead of a Merkle tree to pack the transactions into the Block more efficiently. The Tri tree structure allows data retrieval faster than Merkle tree. At the same time, the Tri tree can serve the purpose of Merkle tree which generates a single hash code as a root to be inserted into the Block's header so that the Block does not need to hold the entire contents while being propagated throughout BC peer to peer network.

(c) The incorporation of a secure payment protocol where the Patient Centric Agent and Miners have an account with traditional banks to purchase virtual credits using conventional currency. The payment protocol improves the system 's sustainability.

Figure 3.2 displays Blockchain components that we have customised to reduce energy consumption and increase throughput. The upper left part of Figure 3.2 shows a conventional Blockchain and the upper right of Figure 3.2 depicts the modification of our customised private Blockchain for the consensus process. The lower left part presents the method of packing transactions in Block for ensuring their integrity in a conventional Blockchain and the bottom right part presents the method of organizing transactions in our customized Blockchain.

The core components of our customized Blockchain are derived from the Bitcoin's Blockchain by optimizing them with respect to power consumption and throughput. However, the PCA can be incorporated in Ethereum network and can utilize smart contracts to interact with Ethereum Blockchain. Therefore, the term "Ethereum" is added in Figure 3.2.

Finally, the performances of the fundamental algorithms in the framework were analyzed by designing a private Blockchain using simulators. The high-level view of the IoT eHealth framework is illustrated in Figure 3.3. The architecture presented in Figure 3.3 includes the Patient-Centric Agent between Sensor Data Processing component in the patient end and Blockchain. The PCA has three main features: Data Management Module (DMM) to decide the storage medium of health data, Miner Management Module (MMM) to control and manage a customized Blockchain and Security Service Module (SSM) to perform authentication, control access, and generate dynamic security keys.

Some features of this architecture such as transactions management, data privacy, registration and identity can be handled using smart contract on private or public/private Ethereum Blockchain. However, for other features such as key generation, data management, access control, and consensus management, designing and implementing own Blockchain is required for assessing performances.

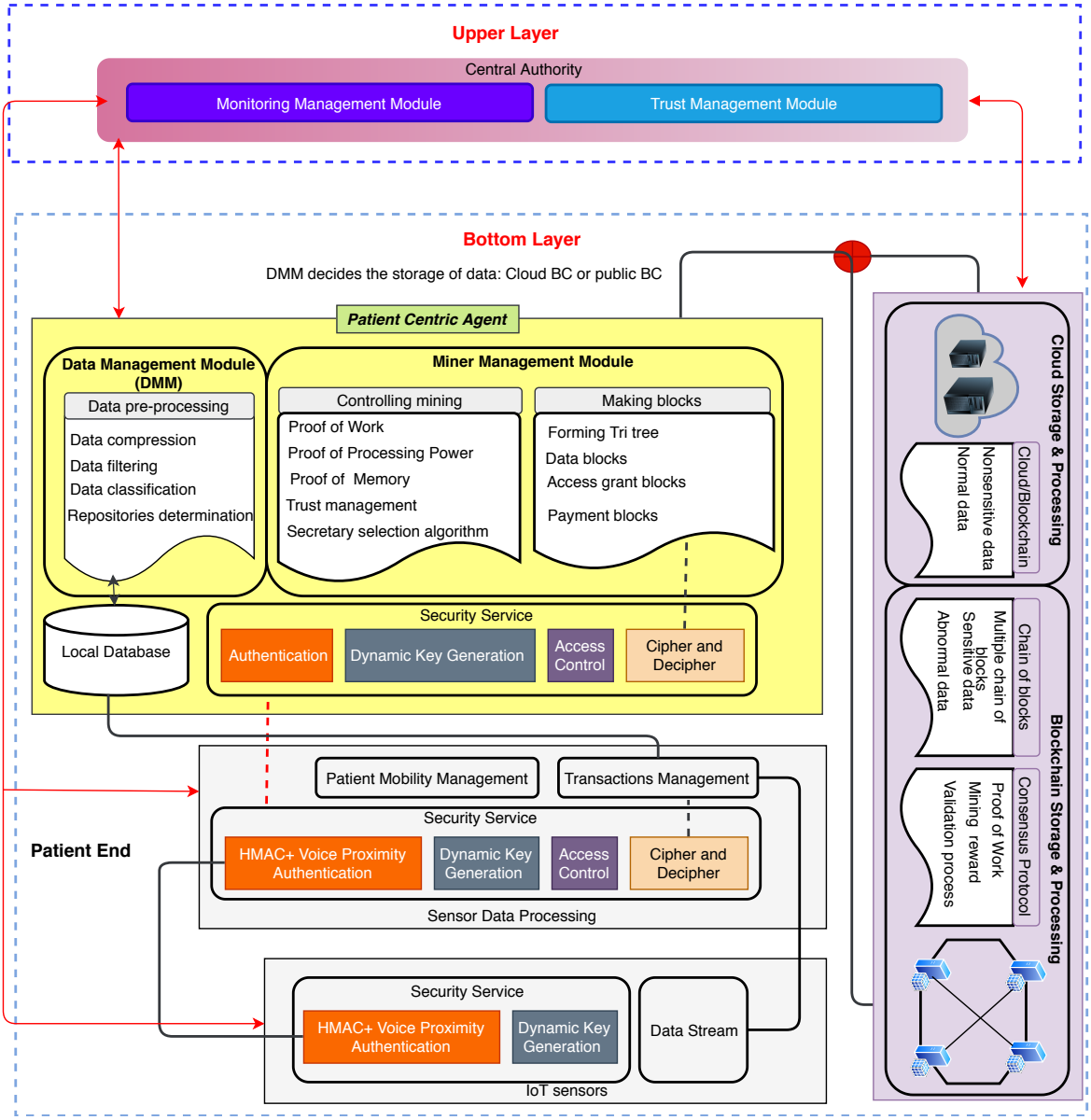


Figure 3.3: The high-level view of the PCA managed IoT eHealth Architecture.

The contents below of this chapter were published in the journal of IEEE Access, IEEE in June 2018. The current impact factor of the journal is 3.75. The article has already been cited by 110 times (according to Scholar Google).
M. A. Uddin, A. Stranieri, I. Gondal and V. Balasubramanian, "Continuous Patient Monitoring With a Patient Centric Agent: A Block Architecture," IEEE Access, IEEE, vol. 6, pp. 32700-32726, 2018. doi: 10.1109/ACCESS.2018.2846779

Abstract

The Internet of Things (IoT) has facilitated services without human intervention for a wide range of applications including continuous Remote Patient Monitoring (RPM). However, the complexity of RPM architectures, the size of datasets generated and limited power capacity of devices make RPM challenging. In this chapter, we propose a tier based End-to-End architecture for continuous patient monitoring that has a Patient Centric Agent (PCA) as its center piece. The PCA manages a Blockchain component to preserve privacy when data streaming from body area sensors needs to be stored securely. The PCA based architecture includes a lightweight communication protocol to enforce security of data through different segments of a continuous, real time patient monitoring architecture. The architecture includes the insertion of data into a personal Blockchain to facilitate data sharing amongst healthcare professionals and integration into electronic health records while ensuring privacy is maintained. The Blockchain is customized for RPM with modifications that include having the PCA select a Miner to reduce computational effort, enabling the PCA to manage multiple Blockchains for the same patient, and the modification of each block with a prefix tree to minimize energy consumption and incorporate secure transaction payments. Simulation results demonstrate that security and privacy can be enhanced in Remote Patient Monitoring with the PCA based End to End architecture.

3.1 Introduction

Internet of Things(IoT)[57] applications in the modern healthcare system include devices, services and wireless sensors that detect physiological signs with wearable or ingestible sensors[58] that stream data to remote, and often Cloud based servers. Secure continuous monitoring of patient's physiological signs[59] has the potential to augment traditional medical practice, particularly in developing countries that have a shortage of healthcare professionals[332],[333],[334].

Remote Patient Monitoring(RPM)[61] involves the integration of physiological data collected with Wearable or Implantable Medical Devices(IMDs), with other data including demographic, health record and geographic location data.

The challenges of designing effective, efficient and secure remote patient monitoring systems include the aggregation and indexing of huge streams of continuous data while maintaining patient privacy. Privacy[335] refers to one's personal space that also includes the capacity to have control over data and determine access levels to be granted to others.

In 2013[336], an alarming 44 percent of all registered data in targeted medical companies was breached. Data breaches reportedly increased by 60 percent from 2013 to 2014 which led to financial losses that increased by a remarkable 282 percent. The vulnerability of wireless communications and relatively weaker cryptographic techniques compared to wired communications make RPM communications an easy target[337].

The threats to the confidentiality, integrity, and availability of healthcare information come from insiders, and outsiders in addition to operational environments[338]. Insiders such as healthcare professionals and support staff, service providers, and outsiders such as hackers threaten health information security by gaining unauthorized access to confidential data. Actions by unauthorized persons can result in alteration of patient's information and can even cause death. Breaches of privacy can erode trust that patients and health care professionals place in the system.

Software disruption caused by viruses, worms, and malware, in addition to resource misuse such as personal use of systems can also threaten health information systems. Communication infiltration, interception, embedded malicious code and repudiation of patient's data pose threats to the confidentiality and integrity of patient's data. Accidental misrouting, technical infrastructure failure, and operational errors can also jeopardize the security of health information.

As outlined in the next section, existing RPM architectures are yet to address these threats. Therefore, there is a need for architectures that afford greater protection of RPM devices and software against attacks.

An eHealth framework [339] for RPM requires that privacy should be preserved while enabling access by authorized healthcare professionals.

Efforts to ensure privacy in RPM have been made in recent years however most approaches focus on a single link in the architecture that chains data from patient sensors to health care professionals through intermediary devices and servers. The archetypal RPM architecture involves sensors near, on or in the patient transmitting data wirelessly using Bluetooth, ZigBee or customized protocols, through a Body Area Wireless Sensor Network to a Base station that initially process data and transmits it to remote servers for further processing.

An effective and efficient RPM needs to address issues of rapid storage at appropriate security levels, user authentication, access control, mobility management and sustainability of patient's health data.

In this article, an End-to-End eHealthcare architecture is advanced that addresses RPM healthcare data management issues to ensure that appropriate levels of trade-off between effectiveness and privacy can be established for rapid, secure data storage and access, user authentication, role based access, and sustainability . Key features of the architecture include a Patient Centric Agent to co-ordinate End-to-End data streams and a Blockchain component for distributed storage of parts of the data. A Patient Centric Agent, described below is proposed to perform the role of determining the data storage security level required. The PCA performs four main functions:-

1. **Rapid Storage and Access:** Large volumes of data streams can be generated by body area sensors. Some conditions such as a sudden arrhythmia may demand a quick response[340]. However, the analysis of a patient's data to determine appropriate actions as data rapidly streams in from sensors challenges computational processing and storage resources. Further, not all of the data generated from patient's body sensors can be assumed to be sensitive or required to be stored[341]. Some raw data streams may even be replaced with simple descriptors over aggregated data such as "normal heart rate pattern" for 24 hours or the result of real time analyses such as those proposed[342], [343],[344],[345]. Chung et al.[342] defined a threshold of separating normal and abnormal ECG data. The ECG signals which width is less than 100 ms and the R-to-R interval is between 0.8s and 0.9s or width less than 60ms, and the R-to-R interval goes beyond 1.1s are classified as Normal ECG waveform. The signals that do not fall in this range are classified as abnormal ECG data. Other streamed data can be presumed to need to be stored but without strong encryption. This means that a process is required to rapidly determine and execute a Data Storage Security Level required

for a stream of data.

2. **User Authentication:** Only authenticated nodes should participate in the transfer of data to prevent attackers intercepting data flows. Although asymmetric key cryptography algorithms ensure data cannot practically be decrypted without the key, they cannot guarantee the owner of a public key is the legitimate owner without the involvement of a trusted authority to issue public/private key pairs. But reliance on trusted authorities for key management results in security, fault tolerance and bottleneck problems in IoT settings[346]. In RPM, a compromised trusted authority might stop the real-time monitoring of all of the patient's physiological data. Further, asymmetric key cryptography[337] is computationally expensive and places great demands on power constrained battery operated devices essential in RPM architectures. The key management is performed by the PCA as a Trusted center at patient's end. The PCA uses symmetric key cryptography for BSN and asymmetric cryptography for SDP and customized Blockchain.

3. **Role Based Access:** Role Based Access Control(RBA) [347] refers to restricted privileges of healthcare professionals on patient data according to their expertise or roles. A patient's physiological data may contain sensitive information where a patient prefers access to be restricted to selected physicians. Inappropriate interpretation of health data may lead to incorrect treatment that might eventually result in long term consequences for a patient's health. The PCA ensures Role Based Access using Access Grant Transaction of the Blockchain.

4. **Sustainability:** A financial model is required to ensure that all participants in the electronic health data storage service receive appropriate incentives to ensure the eHealth systems remain sustainable, and affordable. Financial transactions between healthcare professionals, patients, insurers, government and others are commonplace. So a secured fund transaction process is required to be included in RPM architecture.

The framework advanced in this article, includes Blockchain technology embedded into an End to End architecture. Blockchain for cryptocurrencies is a shared, authenticated, auditable and tamper-proof distributed database[73]. The anonymous properties of a transaction in digital currency address some challenges inherent in patient privacy. But existing Blockchain technology in digital currency cannot be applied as is, to IoT based RPM data because of high computational costs and long transaction processing times. RPM data can stream from sensors so rapidly that it cannot be feasibly processed and added to a Blockchain in real time resulting in delays might discourage patients from using Blockchain. Volunteer miners might also be reluctant to join Blockchain networks owing to the large storage and processing requirements.

We propose that RPM challenges can be reduced with the inclusion of a Patient Centric Agent(PCA). The Patient Centric Agent(PCA) has oversight over the End to End flow of data. The PCA determines the storage, security and access level required at any point in time. The PCA coordinates different segments of RPM such as patient sensors and devices, Blockchain nodes, and healthcare service provider devices. The PCA determines whether a stream of data should be stored in a Blockchain and manages the process, if so. The PCA executes on a machine with mass memory capacity and high processing power. No studies to date have advanced the notion of embedding Patient Centric Agent with customized Blockchain for RPM.

The PCA based End-to-End RPM architecture securely connects a patient's BSN to healthcare providers through different intermediate devices. It has the following design elements, explained in more detail throughout the article;

1. Two tiers-One tier deals with the stream and storage of data. The second tier, called the Healthcare Control Unit, deals with auditing and key management.

2. A secure communication protocol from BSN to patient's smartphone and smartphone to the Patient Centric Agent(PCA). This involves a lightweight authentication algorithm that includes dynamically generated sessional symmetric keys to confirm an End to End security as well as consumption of less power.

3. A Blockchain customized for RPM. Modifications include:

    - The task of selecting a Miner is left to the PCA so that computational effort is reduced and multiple Blockchains can be accommodated.
    - The modification of a block with prefix tree to minimize energy consumption and incorporate secure transaction payments.

The architecture advanced here envisages the PCA playing a central role enforcing security, mediating access to relevant electronic health records, storing particularly sensitive RPM data in a distributed manner, and enabling secure payments.

In the paper, we review related papers in Section 7.2and describe our proposed framework in Section7.3. The performance of key algorithms in the architecture is demonstrated in Section 6.4 before concluding remarks.

3.2 Related Work

We review the state-of-the-art works in three categories: traditional RPM solutions, attribute based RPM solutions and Blockchain based RPM solutions.

3.2.1 Conventional RPM Solutions

Codeblue[348] is one of the earliest healthcare architectures developed based on BSN worn by the patient. Medical sensors wirelessly transmit the sensing data to end users such as PDA(Personal Digital Assistance), laptop, and personal computer. Healthcare professionals issue queries for the analysis of patient's data in a publish-subscribe manner. Although authors in the Codeblue project highlighted the need for security and privacy with medical data, they did not include privacy and security protection in their architecture.

Alarm-net[349] is a heterogeneous network architecture consisting of body sensor networks and environmental sensors for patient health monitoring in the assisted living and home environment. The circadian activity rhythms module in Alarm-net help to adjust context-aware power management and privacy policies. Alarm-net which is connected to BSN, back-end server and IP network imposed some network and data security policies for physiological, environment, behavioral parameters about residents. However, Wood et al.[349],[350] showed that Alarm-net could not guard against the leakage of resident's location. Further, Kumar et al.[351] pointed that hardware built in cryptosystem in Alarm-net makes the application highly platform dependent. Although Alarm-net performs some initial analysis on sensor data for power management, it did not focus on storage management, patient mobility management, security level of streamed physiological data as our proposal extend these techniques.

UbiMon[352] proposed by Ng et al. is BSN based healthcare system and addresses the issues of wearable and implantable sensors for distributed monitoring. Although UbiMon is an ubiquitous healthcare architecture consisting of LUP(Local Processing Unit) that can detect patient's abnormalities and issue instant warning to healhcare service provider, central server and work station for physician, but Ng did not consider security and privacy in their ubiquitous healthcare monitoring architecture. We extended this architecture by placing a smart agent at the patient's end and replace the central server with Blockchain technology. We also embed proper security at each segment of our architecture. Intelligent analysis of huge streamed physiological data requires a dedicated patient agent at the patient's end.

Chakravorty designed a wide-area mobile patient monitoring called MobiCare [353] to collect physiological conditions of patients continuously. MobiCare improves the quality-of-patient care and Mobicare server gives access to physiological data off line to medical staff. Although Chakravorty addressed the security and privacy for real-time applications, secure localization, and anonymity are not yet implemented.

Sensor Network for Assessment of Patients(SNAP)[354] has been proposed to address security concerning wireless health monitoring but it does not authenticate users while providing medical data. Furthermore, adversaries can intercept or modify physiological data because text to the controller is not encrypted in the architecture.

MEDiSN[355] consists of multiple physiological motes powered by a battery for in-hospital patient monitoring. The MEDiSN architecture addresses the issue of reliable communication, routing, data rate, and QoS. In the design, authors expressed the necessity of encrypting physiological monitoring data, but their study did not report the encryption technique that ensures confidentiality and integrity.

Moosavi[356] proposed an End-to-End security scheme for mobility enabled healthcare IoT. The End-to-End Security Scheme architecture consists of three-layers, the device layer(BSN), fog layer(gateways, network router), and Cloud layer. Moosavi proposed a secure and efficient end-user authentication and authorization architecture based on the certificate DTLS(Datagram Transport Layer Security) handshake, secure end-to-end communication based on session resumption, and robust mobility based on interconnected smart gateways. DTLS depends on a trust center and involves a higher number of flights to complete the authentication process. Storage of physiological data in the Cloud causes higher latency and consumes bandwidth for continually monitoring healthcare system. Furthermore, Moosavi et al. did not focus on access control on patient's data in the Cloud. We advance our End-to-End e-healthcare architecture incorporating Blockchain technology, which includes a secure payment system and employ more lightweight authentication at the patient's end. A trusted authority in our architecture has a role to play in certifying healthcare professionals through Blockchain.

Gope[339] proposed a modern healthcare system called BSN-Care for RPM. BSN-Care architecture includes conventional devices such as BSN, Local Processing Unit(LPU) and BSN-Care Server in healthcare monitoring system. In BSN-Care, BSN-Care Servers analyze patient physiological data transferred by LPU using heuristic approach. BSN-Care Server alerts healthcare givers if physiological data exceed a preset thresholds. Gope proposed a one-way hash based lightweight authentication method that uses shadow identity to preserve patient privacy and security. Data processing may bottleneck because the architecture depends on a single server. Yeh [357] also used the architecture of BSN-Care. He proposed hash based public/private key authentication LPU to BSN-Care Server and a hash based lightweight authentication integrating GPS for BSN to LPU.

In this work, we extend the BSN-Care lightweight authentication by including proximity and HMAC[358](keyed-Hash Message Authentication Code) for Body Area Sensor Network to smart-

phone communication channel. Proximity helps BSN devices in RPM to detect attacker's devices because of their physical location. In proximity authentication, two entities estimate distances between each other by exchanging some signals such as radio or voice. Therefore, device cannot claim incorrect physical location during authentication. We include GPS and options for using different kinds of encryption algorithms for smartphone to PCA communication channel. Spoof attack can be prevented by GPS. Option on using different encryption algorithm along a communication channel delays the attacker's effort to break data confidentiality.

Central Server based architecture [348]-[357] serves to store and analyze health records of limited number of patients. Therefore, we propose a scalable healthcare architecture integrating customized Blockchain, which is scalable and robust against attacks.

3.2.2 Attribute Based RPM Solutions

Attribute based authentication[359] refers to validating entities/persons on conditions that they possesses a certain number of attributes such as a person must be a doctor, heart specialist and service experience of 10 years to access a patient's record with heart diseases. A trust party ensures that a person owns the required properties in attribute based authentication and encryption. Two attribute-oriented authentication and transmission schemes for secure and privacy-preserving health information sharing in health social networks (HSNs) that is a social platform like Twitter for patients and healthcare service providers to share medical records and their views are proposed in[360] where the access policy is defined by a target set of attributes such as patient identity, diseases history, and social status. Only users who satisfy the access policy are able to decrypt the cipher text. Privacy is preserved in this approach because it does not require the identity of the entity. They demonstrate that the proposed schemes can effectively resist various attacks including forgery attack, attribute-trace attack, eavesdropping attack, and collusion attack. However, authors didn't focus on revocation that refers to remove the access capabilities of authorized users any time and write operation on attribute-based encrypted medical data. Lounis [361] pointed that medical data encrypted on attributes in the Cloud needs to be downloaded and stored again if the access policy associated with attributes changes. The computational cost of attribute based cryptosystems increases linearly with the number of attributes. The papers used symmetric key to encrypt data files in Cloud. The symmetric key is encrypted with access policies associated with attributes. Therefore, symmetric key requires to be encrypted if access policy is changed.

Liang[362] also used patient's attributes to ensure authentication in smart home based pervasive healthcare system while communicating to online healthcare provider. To preserve a patient's location privacy, a receiver chain is formed where the source node requests a neighbor node to be a proxy source to enable vendor-to-patient communication. However, the source node can still be traced along the chain. Li[363] proposed a scalable personal health records to be stored in the Cloud using attribute based encryption so that patient's record can be securely shared with multi-authority by maintaining appropriate privacy. The scheme also supports the dynamic modification of attribute policies, on-demand user/attribute revocation and break-glass access at the time of emergencies. Although attribute based encryption schemes provides security and privacy of patient's record, the scheme is not lightweight enough to implement in wearable medical sensors and smartphone[364].

3.2.3 Blockchain based RPM Solutions

In the End to End frameworks proposed by[348]-[363], the patient must depend on Trusted Centers for key management. The devices that streamed real-time data in RPM experienced higher latency and communication overhead to obtain keys from Trusted Centers. Further, these frameworks don't ensure the availability of patient's data if the traditional server or Cloud server is compromised. Single point health applications described by[79] have some drawbacks such as when a user goes to another hospital, the previous hospital may be reluctant to share data. Further, healthcare professionals can violate privacy by providing patient's data to a third party. Health data might suffer from a single point of failure.

In addition, traditional remote patient monitoring system requires authorization between remote end user/healthcare center and medical devices at the patient's end. The authorization causes communication overhead (required bandwidth, computational overhead, the number of transmitted message).

Blockchain healthcare architecture reduces the communication overhead to eliminate the requirements of running authorization algorithm for the remote end user to access data from the Blockchain[66], [365]. Further, Blockchain healthcare provides the facilities of peer-to-peer record's transmission without the involvement of third party trust, interpretability of longitudinal healthcare records, transparency with pseudonymity and irreversibility of records. Patients in Blockchain have options to hide their identity with alphanumeric address or show proof of their identity to others[66].

On the other hand, Blockchain technology applied in RPM involves a high-cost consensus process, IoT data can plausibly be generated faster than Blockchain consensus approach can validate the proof of concept[366]. An optimized Blockchain is required in order to preserve privacy in IoT based remote patient monitoring.

Zhang et al.[76] introduced a modified IEEE 802.15.6 authentication association protocol by considering the limited processing power of sensor nodes for pervasive social network based healthcare between BSN and smartphone. IEEE 802.15.6 requires two scalar multiplications at the BSN end and two scalar multiplications at the smartphone. In modified 802.15.6, sensor device performs one scalar multiplication. Secondly, they use a Blockchain oriented architecture that consists of the body area wireless sensor network and PSN(powerful computer, laptop, and smartphone build this network). They included a coordinator node(smartphone) placed at user end, which broadcasts transaction among PSN to verify its signature(using the master key exchanged through the modified protocol described in the first part of their works) of the sensor and the node itself. The transaction of Blockchain in this proposal doesn't contain physiological health record. The transaction includes patient or healthcare providers' meta data such as identity, address, and diseases. Their modified IEEE 802.15.6 still involves high computation for BSN because scalar multiplication is computationally expensive operation.

Bowhead [78] is a Blockchain based healthcare application. The application introduces some medical devices such as test cartridge, test reader and dispensing devices to capture patient data. They have also designed an App that prompts patients about the dosage and time of taking medication. The user can provide his information to Bowhead's application through different body area medical sensors and the application stores that information to a Blockchain based database. Although white paper describes the procedure for collecting patients data, it does not describe how the stream of medical data produced from patients body fit to the existing Blockchain.

MeDShare[75] is also a medical data sharing system among Cloud service provider via Blockchain contract. The contract refers to a program written by user defining terms and conditions of an

agreement. The Blockchain nodes only justify the rules of the agreement. The authors propose an architecture for sharing documents among requestors. In design approach, they discussed the system setup, requested file, package delivery, auditing and provenance in detail where the function of each layer of their architecture and smart contact technology in Blockchain are integrated to share data securely. However, MedShare constitutes only a sub-system in RPM architecture.

Health Care Data Gateway(HDG)[77] is a smartphone based App that integrates traditional database and Blockchain distributed database to manage patient health data. They proposed a multiparty computation(MPC) approach where third parties can access data but not alter data. HDG consists of three layers called Storage Layer, Data Management Layer, and Data Usage Layer. Cloud is the platform for Storage Layer in Blockchain fashion. The Data Management Layer comprises individual devices. All kinds of request either incoming or outgoing will pass through this layer. The Data Management Layer helps in indexing and making queries to retrieve data from the Cloud. Data Usage Layer includes physicians, electronic medical record system, and data analytical algorithms. The diagnostic center will directly send data to patients and patients transfer authority for the data to the doctor for further analysis. The third party might continue analyzing data without accessing user's data through MPC in the Blockchain. They proposed a unified scheme to store medical data. The paper does not mention how consensus mechanism and auditing processing work in Cloud based Blockchain.

The proposal might be integrated with the body area sensor network. Yue et al. [77] do not show how patient record can be encrypted or accessed at granular level and how Blockchain tackles stream of medical sensor data. In IoT healthcare, one key for all the users in the Blockchain raises privacy risk and one key for every individual involved in the Block chain is also not feasible because user might be still identified through inspection and analysis of available open information on the Blockchain[367]. Moreover, the key should vary for every chunk of data block in the Blockchain.

Zhao et al. [79] proposed a fuzzy vault based key management in Blockchain health architecture. The architecture of the Blockchain based health framework includes wearable sensor nodes on the patient's body, some implanted nodes and gateway nodes for the body area sensor networks. The Gateway collects physiological data from wearable sensor nodes and sends aggregated data to some pointed hospital which individually make a block in Blockchain and message generated from the Gateway is considered as a single block. Wearable sensor nodes produce a key before sending physiological data to the Gateway node and encrypt the data with the key generated from the signal of patient's body. Blockchain communities or healthcare professionals cannot leak the patient information. The patient can only recover the key from her physiological data to decrypt data. However, this approach causes significant burden for power constrained medical sensors because these sensors require to construct the key from patients' physiological data during decryption.

Linn et al.[80] planned an off Blockchain approach for health data storage called Data Lake and a Blockchain containing all authorization transactions. Data in an encrypted format is stored as key-value pairs in the data lake. The user, health data provider, and doctor use mobile apps to access data through Blockchain from the data lake. After completing analyses such as different types of medical tests, the provider inserts a signature and authenticates a user to access that result through Blockchain. In the same way, the user can offer authentication and authorization of its data to a doctor through Blockchain. They do not show the design elements of Blockchain and off Blockchain does not provide the access to the medical record that healthcare providers need.

In eHealth frameworks, the mechanism to provide the patient's data to health professionals is not highlighted in central server based architecture. On the other hand, streamed data is generated from the medical sensors in continuous patient monitoring system, the current architecture of

Blockchain based healthcare also overlooks the efficient processing of huge stream of patient's data so that patients get a rapid response from healthcare professionals. There is still a need to design an End-to-End eHealthcare framework merging Blockchain with legacy healthcare architecture.

3.3 Proposed Secure Patient Monitoring Architecture

The proposed architecture comprises two tiers; the lower tier provides the data streaming and storage solution whereas the upper tier manages keys healthcare provider and is called Healthcare Control Unit(HCU). The lower tier includes six systems illustrated in Figure 7.2. Body Area Sensor Network(BSN), Sensor Data Provider(SDP) such as smartphone, Patient Centric Agent(PCA), Blockchain, Healthcare Provider Agent(HPA) and Healthcare Provider's Wallet(HPW). In Figure 7.2, BSN is connected to Patient Centric Agent(PCA) through Sensor Data Provider such as a smartphone. PCA is connected to Blockchain network, Cloud and the Healthcare Control Unit. Healthcare Provider Agent connects Blockchain, Healthcare Control Unit and Healthcare Wallet at healthcare provider end. The architecture is explained in accordance to the communication links between different segments below and the functional view of the architecture is depicted in Fig. 7.4. The architecture is designed to scale to large numbers of patients.

There are two parts of the architecture: Healthcare Control Unit and Data Streaming and Storage. In Fig. 7.2, we used the term "Tier" to refer to layers.

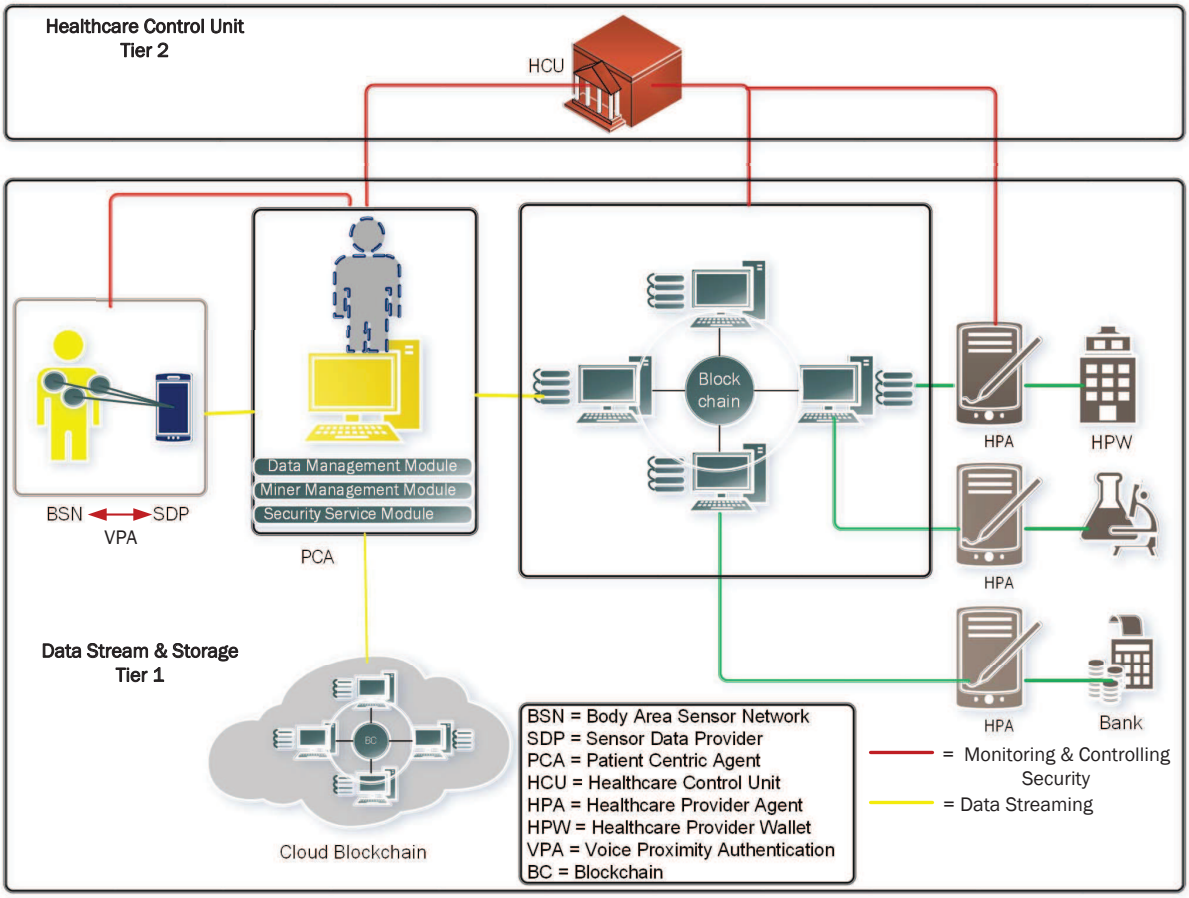


Figure 3.4: The tier based Remote Patient Monitoring architecture.

3.3.1 Body Area Sensor to Sensor Data Provider

In this section, we discuss BSN and SDP, and mutual authentication process between these two segments.

3.3.1.1 Body Area Sensor Network(BSN)

Different types of wearable sensor devices such as motion tracker, biophysiological sign measurement devices(EEG, ECG, BSC etc.)[368] form the Body Area Sensor Network. In our architecture, we also consider n number of wireless wearable sensor devices measuring physiological signs in the Body Area Sensor Network(BSN). Devices in BSN are extremely power constrained and have very limited processing power[369]. Typically, these devices send patient's data to a nearby smartphone using the Bluetooth or ZigBee protocol[364].

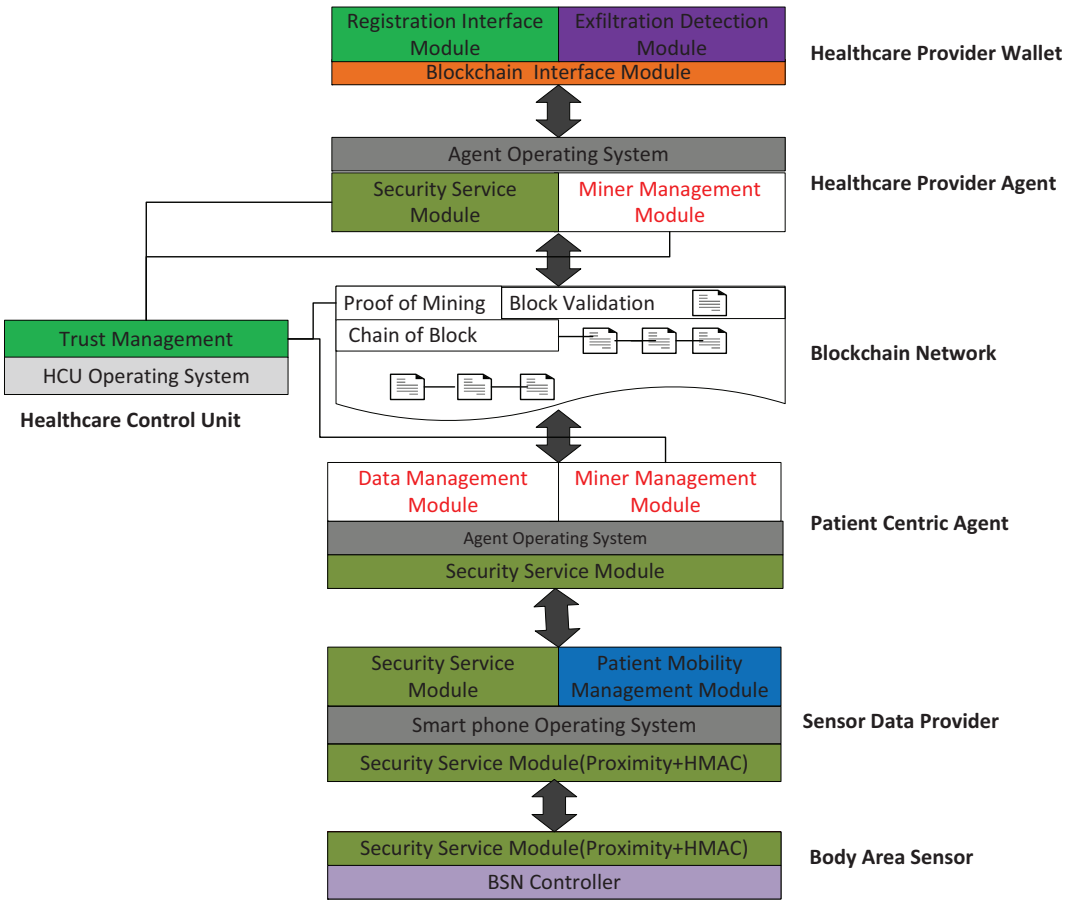


Figure 3.5: Conceptual view of the tier based health monitoring architecture

3.3.1.2 Sensor Data Provider(SDP)

The Sensor Data Provider(SDP) is software that executes on a mobile device, cellphone or modem. We assume that a patient has a dedicated smartphone to receive health data from medical sensors in BSN and wirelessly provides the data to the Patient Centric Agent (PCA) described in 3.3.2. The data stream generated from a sensor is partitioned by the SDP in variable size data frame windows[370]. For instance, the size of the window is set by the SDP for each variable using

simple heuristics and varies according to the fluctuations in data rates. The SDP does not perform any aggregation of data such as packaging heart rate and breath rate streams together. Further, the SDP does not pre-process data to remove outliers or abnormal values.

The SDP includes two components. A Security Service Module(SSM) performs cryptographic operations such as key generation, authentication, encryption and decryption and a Patient Mobility Management Module continues transmitting patient's physiological data to PCA using long range communication standards such as NB-IoT[364] while patient moves to a new place.

Public/Private encryption requires devices having considerable processing power[371]. Wireless sensors in BSN lack the processing power required for public key encryption. Furthermore, confirmation of ownership's legitimacy of public/private key is not feasible without a third party trust center. Although symmetric key authentication is a promising solution for the BSN to SDP segment, key sharing is vulnerable to man in middle attack. According to Malina [337], Proximity based User Authentication(PUA) introduced by[372] can address the key exchange challenges of IoT devices with power and memory limitations.

3.3.1.3 Authentication BSN to SDP

We design a mutual authentication approach by integrating Proximity User Authentication(PUA) and HMAC[358](keyed-Hash Message Authentication Code) for BSN to SDP channel. The mutual authentication is a two way authentication where both entities in the communication link prove their identities to one another. Entities in PUA perform their authentication based on their physical distance. In RPM healthcare system, legitimate medical sensors attached to the patient's body and smartphone in SDP are usually closer than attacker's devices. Therefore, intrusion to patient's medical sensors will not be successful thanks to attacker's position even if it can discover the legitimate device's session key that is used to produce HMAC.

Radio signal based proximity measurement[373],[374] estimates distance between entities in a communication link by exchanging radio signals. Two entities in close proximity to each other can be assumed to transmit stronger radio signals however radio signals can be easily manipulated by attackers[372]. However, voice signal introduced by [373] requires less power than radio signal to measure the proximity between two entities.

Therefore, voice signal is used in the proposed authentication process to estimate the distance between two entities and to prove their legitimate identities. Voice based Proximity(VP) has some limitations. An attacker might play a recording of the voice while person is asleep and the processing of VP in BSN device adds much computational burden on extremely power constraint medical sensor. Therefore, we presume that BSN and SDP device store the legitimate user's voice and have also a voice or audio processing unit.

3.3.1.3.1 Mutual Authentication Protocol The lightweight authentication protocol confirms SDP receives physiological data from the legitimate patient wearing medical sensors. We assume that BSN and SDP devices produce a sessional symmetric key(K_i) for authentication through a dynamic key generation mechanism described in 3.3.2.3. The mutual authentication BSN to SDP is depicted in Figure 7.7. Here, the authentication process starts with the BSN speaking to the SDP. H() represents Blake2 message digestion code and HMAC represents Marvin message authentication code. Pereira et al. [375] showed that Blake2 Hash and Marvin MAC outperform other approaches in IoT devices in terms of speed and energy. HMAC is faster and requires less computational cost in terms of processing power than public/private key pairs. The cryptographic

notions and meaning are illustrated in Table 3.1. The mutual authentication process is described as follows:

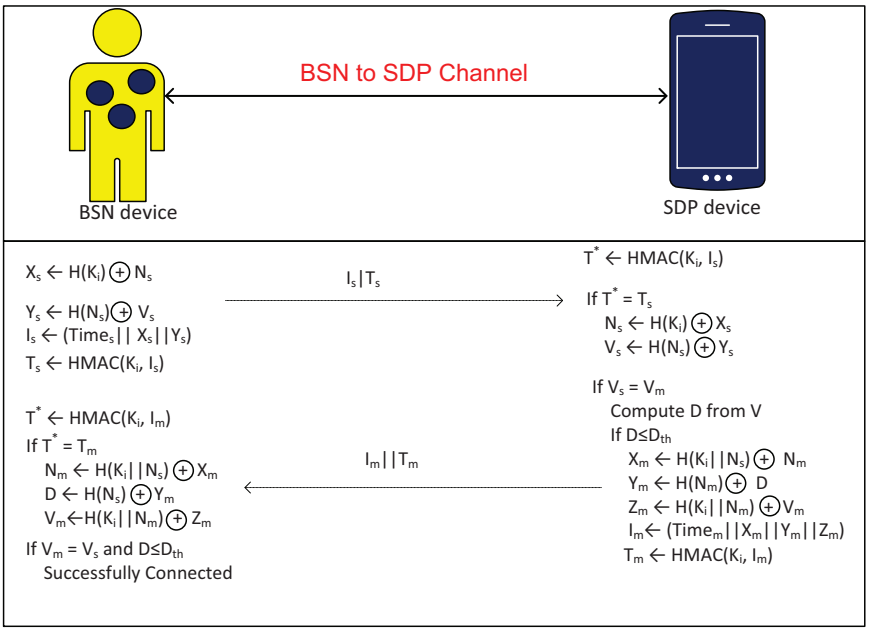


Figure 3.6: Mutual authentication BSN to SDP.

- Firstly, BSN initiates authentication by sending the SDP device a transmission that consists of two messages: An information message consisting of time, nonce, and user voice signal; an HMAC of time, nonce, and voice signal to ensure the integrity of the flight. Here, time, nonce, and user voice signal are encrypted with a one-time pad to hide them from the attacker.

  BSN $\rightarrow$ SDP The BSN device randomly chooses a nonce(N_S) and uses system time($Time_S$) to confirm the freshness of the authentication message. $Time_S$ guards against reply attack and nonce is also used as dynamic identities of BSN and SDP devices during transmission of health data. BSN device performs XOR($\oplus$) operation on the nonce and H() of symmetric key(K_i) to produce $X_S \leftarrow H(K_i) \oplus N_S$ that hides the nonce from attackers. Likewise, Y_S represents one time pad of voice signal and it is produced as $Y_S \leftarrow H(N_S) \oplus V_S$ where V_S represents voice stored in BSN. BSN produces an information message $I_S \leftarrow Time_S || X_S || Y_S$. The attacker might intercept and change information message(I_S). Therefore, BSN computes $T_S \leftarrow HMAC(K_i, I_S)$ where T_S represents the HMAC produced from I_S using a sessional symmetric key(K_i). BSN device sends I_S and T_S to SDP device. Here, T_S ensures that I_S are produced by BSN device.

- Secondly, the SDP device receives the flight from BSN device over an insecure channel and decrypts the information message provided that the verification of HMAC is successful. Next, SDP estimates the distance between the origin of the voice(BSN) and itself if the voice from BSN is identical to SDP's stored voice. After that the SDP also prepares a flight consisting of its information- Time, Nonce, Distance and its Voice, and HMAC of these information using a dynamically generated symmetric key at BSN and SDP end. The SDP also sends its flight to BSN for mutual authentication.

Table 3.1: The cryptography notions and meanings

($\oplus$)	XOR operation
H()	Blake2 Hash operation
HMAC()/MMAC()	Marvin Hash Authentication Code
K_i	Dynamically generated sessional symmetric key
$Time_s$	BSN device's system time
N_s	Nonce generated by BSN device
V_s	User voice/audio stored in BSN device
X_s	One time pad produced from BSN device's Nonce and $H(K_i)$
Y_s	One time pad produced from BSN's voice/audio and $H(N_s)$
I_s	Information message consisting of $Time_s$,X_s,Y_s from BSN
$Time_m$	BSN device's system time
N_m	Nonce generated by SDP device(mobilephone)
V_m	User voice/audio stored in SDP device
X_m	One time pad produced from SDP device's Nonce and $H(K_i\|\|N_s)$
Y_m	One time pad produced from SDP's voice/audio and $H(N_m)$
I_m	Information message consisting of $Time_m$,X_m,Y_m from SDP
$pubKey_m$	Public key of SDP device
$privateKey_m$	Private key of SDP device
$pubKey_a$	Public key of PCA
$privateKey_a$	Private key of PCA
I_a	Encrypted Information Message from PCA

SDP $\rightarrow$ BSN SDP produces $T^* \leftarrow HMAC(K_i, I_s)$ upon receiving I_s and T_s from BSN using the same session symmetric key(K_i). SDP extracts nonce($N_s \leftarrow H(K_i) \oplus X_s$) and voice($V \leftarrow H(N_s) \oplus Y_s$) if $T* = T_s$. SDP measures distance(D)between the origin of the voice and the SDP from voice signal(V_s) provided that $V_s = V_m$(voice stored in SDP device). If distance(D) between BSN and SDP doesn't exceed the threshold distance(D_{th})set by the PCA, then SDP also randomly chooses a nonce(N_m) and computes $X_m \leftarrow H(K_i||N_s) \oplus N_m$, $Y_m \leftarrow H(N_m) \oplus D$, and $Z_m \leftarrow H(K_i||N_m) \oplus V_m$. SDP forms an information message $I_m \leftarrow Time_m||X_m||Y_m||Z_m$ and produces $T_m \leftarrow HMAC(K_i, I_m)$ where HMAC's result of I_m using the symmetric key. SDP device sends I_m and T_m to BSN device.

- Thirdly, the BSN verifies the flight from SDP in a similar fashion.

  BSN $\rightarrow$ SDP BSN verifies T_m and extracts N_m, D and V_m to check if $V_s = V_m$.

3.3.2 Sensor Data Provider to Patient Centric Agent

The Patient Centric Agent that embeds Blockchain with SDP and BSN at the patient's end is discussed in this section. Following that, the mutual authentication process between SDP and

PCA, sessional symmetric key generation and the communication protocol for BSN, SDP and PCA are discussed.

3.3.2.1 Patient Centric Agent(PCA)

The Patient Centric Agent is software that executes on a patient's laptop, desktop or a dedicated server. The patient agent node contains three modules: medical Data Management Module(DMM), Security Service Module(SSM), and Miner Management Module(MMM). The agent node plays a critical role in our architecture and the functionality of each component is described as follows:

- **Data Management Module(DMM):** Continuous patient monitoring system generates huge volumes of data[340]. However, some data is not required to be stored at all, whereas other data requires storage with strong encryption. For instance, unusual heart patterns in cardiovascular patients is likely to be clinically useful and would normally be stored. An intelligent module is needed to determine the level of storage required for each data stream[376]. The module classifies patient's data as Normal, Eventful and Uneventful following[377],[378]. The module stores uneventful data locally and also sends uneventful patient's data to Cloud in case healthcare professionals requires the data. We do not discard any physiological signal of the patient, because even uneventful data may be useful for some future purpose such as research. Further, physiological data compression[379],[380], [381] that requires high computational cost is performed in DMM instead of BSN.

- **Miner Management Module(MMM):** The PCA participates in storing streamed data in a Block. The PCA, through its MMM might also act as a Miner in case no other Miners respond within a certain time. The module runs the Miner Selection Algorithm(MSA). The MMM and Cloud also form a Blockchain containing uneventful data. The module also collects network information such as availability, CPU resources about Miners in Blockchain from Healthcare Control Unit(HCU).

- **Security Service Module(SSM):** The SSM of PCA continuously analyzes the susceptibility of communication channels such as BSN to SDP, SDP to PCA and PCA to BSN to network security attack. The module excludes devices compromised by attacker in BSN and SDP. Further, the SMM periodically sends updated key generation information in BSN and SDP devices. In Blockchain, public/private key is used to hide user identity. Patient's agent might use a set of private/public key. SSM maps a person's public key into one of a few symmetric keys linked to that key and randomly uses one of the linked keys in place of the public key for indexing in Blockchain. Otherwise any Miner can follow a person's public key down the chain to discover all transactions.

3.3.2.2 Authentication SDP to PCA

Proximity authentication is not appropriate for SDP device to PCA because of patient's movement. Here, we include GPS(Global Positioning System)[357] that protects data from spoofing attack [1]. A channel is considered more secure if the channel changes data encryption algorithm for a new session, because attackers do not have knowledge about channel's encryption/decryption mode. So, SDP device and PCA agree on an encryption approach from a predefined algorithm set(Triple

[1] A spoofing attack is performed by the attacker or malicious program by successfully impersonating health data on behalf of patients. ARP, DNS and IP spoofing are some example of spoofing attack.

DES, RSA, Blowfish, Twofish, AES(CBC, CTR, OCB, CCM, GCM)[382]) through authentication. But usage of different kinds of encryption algorithm at BSN to SDP channel is not feasible because of resource constraints on IoT devices in BSN. According to[375], AES-CTR is the most suitable encryption mode among AES, Curupira and Trivium for power constrained IoT devices in terms of speed and energy. BSN to SDP channel uses AES-CTR encryption mode to preserve health data confidentiality.

PCA might use several public/private key pairs. SDP device and PCA are required to validate new public/private key. The session key(K_i) is used to validate new public/private key in the authentication process so that devices don't require third party trusted center to verify public/private key.

3.3.2.2.1 Mutual Authentication Protocol

- Firstly, SDP device and the PCA require validating new public/private key pair using their symmetric key as this involves no third party trusted center to certify public/private key of patient's end's device. SDP initiates the authentication protocol by sending a flight formed by the encrypted public key of SDP using the symmetric key and encrypted HMAC of the encrypted public key using SDP's private key to ensure that attackers have not changed the encrypted text and the SDP has produced the HMAC.

  SDP $\rightarrow$ PCA, we suppose that SDP or PCA has a new public/private key pair. SDP makes two encrypted text: $X_m \leftarrow Enc(K_i, pubKey_m)$ and $P_m \leftarrow Enc(privateKey_m, HMAC(K_i, X_m))$ where $pubKey_m$ and $privateKey_m$ are public and private key of a SDP device(mobile phone). X_m is the encrypted text of SDP device's public key using session key(K_i) and P_m is encrypted text of $HMAC(K_i, X_m)$ using SDP device's private key. SDP device sends the flight($X_m||P_m$) to PCA over an insecure channel. $HMAC(K_i, X_m)$ and P_m ensure that the owner of the public key/private key is legitimate and $X_m$andP_m have not been changed by attackers.

- The PCA receives the flight from SDP and decrypts the public key of SDP using the symmetric key(K_i), and the HMAC using SDP's public key which has been encrypted using SDP's private key. Similarly, the PCA makes a flight packing its encrypted public key using the symmetric key and an encrypted text of HMAC produced from the encrypted public key.

  PCA $\rightarrow$ SDP PCA decrypts X_m to obtain public key of SDP device $pubKey_m \leftarrow Dec(K_i, X_m))$. Next, PCA decrypts P_m to get $P^* \leftarrow Dec(pubKey_m, P_m)$ and verify if $P^* = HMAC(K_i, X_m)$. After that, PCA also produces two encrypted text: $X_a \leftarrow Enc(K_i, pubKey_a)$ and $P_a \leftarrow Enc(privateKey_a, HMAC(K_i, X_a))$ where $pubKey_a$ and $privateKey_a$ are public and private key of the PCA. X_a is encrypted text of PCA's public key using a session key(K_i) and P_a is encrypted text of $HMAC(K_i, X_a)$ PCA's private key. PCA sends the flight($X_a||P_a$) to SDP device over an insecure channel. $HMAC(K_i, X_a)$ and P_a ensure that the owner of the public key/private key is legitimate and X_a and P_a has not been changed by attackers as well.

- Secondly, SDP prepares an information message that contains Time, Nonce, Data Encryption Algorithm, Location. SDP first encrypts the information message using PCA's public key, and then ciphertext is again encrypted by using the symmetric key. Two-time encryption ensure that only the legitimate PCA can decrypt the final ciphertext. The SDP produces HMAC of the ciphertext of information message and encrypts HMAC twice; first uses its

private key and then public key of the PCA. The SDP transfers PCA the flight having encrypted information message and HMAC. Encryption of authentication message using both symmetric key and public/private key makes sure that an attacker cannot break the security of the authentication process without knowing both types of key.

SDP $\rightarrow$ PCA The SDP(mobile phone) randomly chooses a nonce(N_m). SDP produces two encrypted text: $I_m \leftarrow Enc(K_i, Enc(pubKey_a, Time_m||N_m$ $||EA||L_m))$ that contains information of time($Time_m$), nonce(N_m), data encryption algorithm (EA), GPS location(L_m) of the SDP device and $T_m \leftarrow Enc(pubKey_a, Enc(privateKey_m,$ $HMAC(K_i, I_m)))$ where $PubKey_m$ denotes the public key of the SDP device. First encryption in T_m using private key of SDP private key ensures that encryption is done by SDP device and second encryption in T_m using public key of PCA ensures that only PCA can decrypt and verify the encrypted text.

- The PCA decrypts an information message using its symmetric key and private key respectively. Next, it verifies the HMAC of ciphertext of information message by decrypting it using the private key of PCA and the public key of SDP as shown in flight 3 of Fig. 7.8. The PCA also prepares a flight with it's information message and HMAC of ciphertext of the information message.

  PCA $\rightarrow$ SDP PCA decrypts T_m to get $T^* \leftarrow Dec(pubKey_m, Dec(privateKey_a, T_m))$ upon receiving the flight($I_m||T_m$) from SDP device. If $T^* = HMAC(K_i, I_m)$, the PCA decrypts information $I_m(I \leftarrow Dec(privateKey_a, Dec(K_i, I_m)))$ using a session key(K_i) and its private key($privateKey_a$) respectively. Here, attackers can only decrypt information if both session key and private key of public/private key pair are known to attackers. After that, the PCA randomly chooses a nonce(N_a) and then PCA produces two encrypted text: $I_a \leftarrow$ $Enc(K_i, Enc(pubKey_m, Time_a||N_a||EA||L_a))$ that contains information of time($Time_a$), nonce(N_a), data encryption algorithm (EA), GPS location(L_a) of the PCA and $T_a \leftarrow Enc(pubKey_m,$ $Enc(privateKey_a, HMAC(K_i, I_a)))$ where $PubKey_a$ denotes the public key of the PCA. First encryption in T_a using private key of PCA's private key ensures that encryption is done by PCA and second encryption in T_a using public key of SDP device ensures that only legitimate SDP device can decrypt and verify the encrypted text.

- SDP verifies T_a and decrypts I_a to obtain PCA's information.

In Figure 7.8, first two flights(1 & 2) between SDP and PCA occur to validate new public/private key and second two flights(3 & 4) represents sharing authentication information.

3.3.2.3 Sessional Symmetric Key Generation

The exchange of symmetric keys for a session is vulnerable to man in middle attack and also causes higher communication overhead. BSN, SDP and Patient Centric Agent in our RPM architecture generate the same session key to reduce communication overhead and security vulnerability before performing authentication based on some pre-shared information by PCA. This means that devices at the patient's end do not require a key exchange mechanism for every new session.

Session Key generation Method: The approach advanced includes a primary secret key(PSK), a linear feedback shift register(LFSR) used by[383] sequence generator with feedback polynomial $f(x_1, x_2, ..., x_m)$, and a hash table(T) that holds random numbers to generate the session key. The session key generation process is explained as follows:

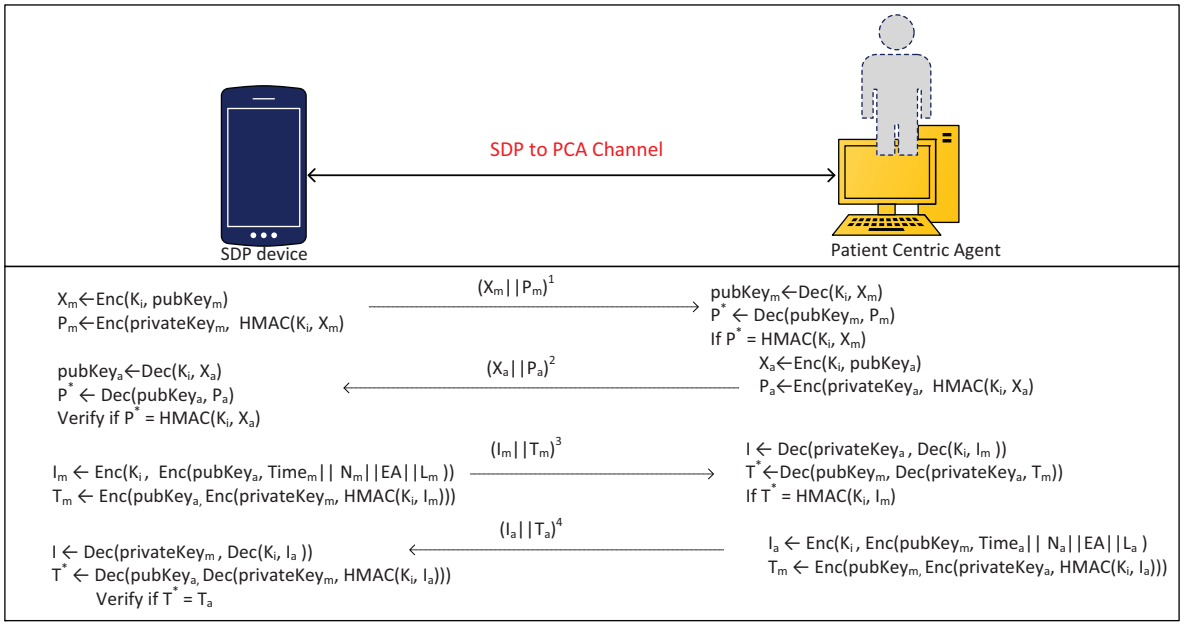


Figure 3.7: The mutual authentication SDP to PCA.

- **Step 1:** Device at patient's end performs MMAC(Marvin Message Authentication Code) operation on the XOR(⊕) operation of linear feedback shift register, and previously used session key using the primary secret key(PSK). Marvin Message Authentication Code(MMAC) that has the best performance in terms of energy and speed in IoT devices is used as MAC operation[375].

- **Step 2:** Another MMAC operation is done on a random number taken from a preshared random number table using the primary secret key(PSK). Later, a sessional symmetric key is generated by performing XOR(⊕) operation on the two MMAC results; the previous MMAC from **Step 1** and this MMAC. K_i = MMAC(PSK, $f(x_1, x_2, x_3, ..., x_n) \oplus K_{i-1}) \oplus$ MMAC(PSK, r_i)

  Where PSK is the primary secret key, K_{i-1} and K_i represents the current and previous session key respectively and r_i is a random value from a table(T) and r_i = T[i mod n] where n is the total number of random number in table(T) and i represents the i^{th} session. Here, the random number creates immunity against rainbow table attack[2] that refers to 2^n input and output pairs pre-computed and stored in a table[384].

- **Step 3:** The hash table(T) containing random numbers is updated by applying the H() on the value of the table repeatedly if all of the random numbers have been used up.

The session key generation algorithm illustrated in Algorithm 1 and in Figure 7.3 where the XOR operation is performed on the output of LFSR which input bit is XOR of the previous state, and the previously used session key. Next, MMAC operation is performed on the XOR output using the primary secret key. Finally, session key is generated from the XOR operation of this MMAC result and MMAC result from random value from the Table(T).

[2]A rainbow table attack makes use of a large database that holds a large number of a hash function's input and corresponding outputs. Attacker stores plaintext and the corresponding hash of plaintext in a table to avoid generation of the hash again during looking up the hash next time.

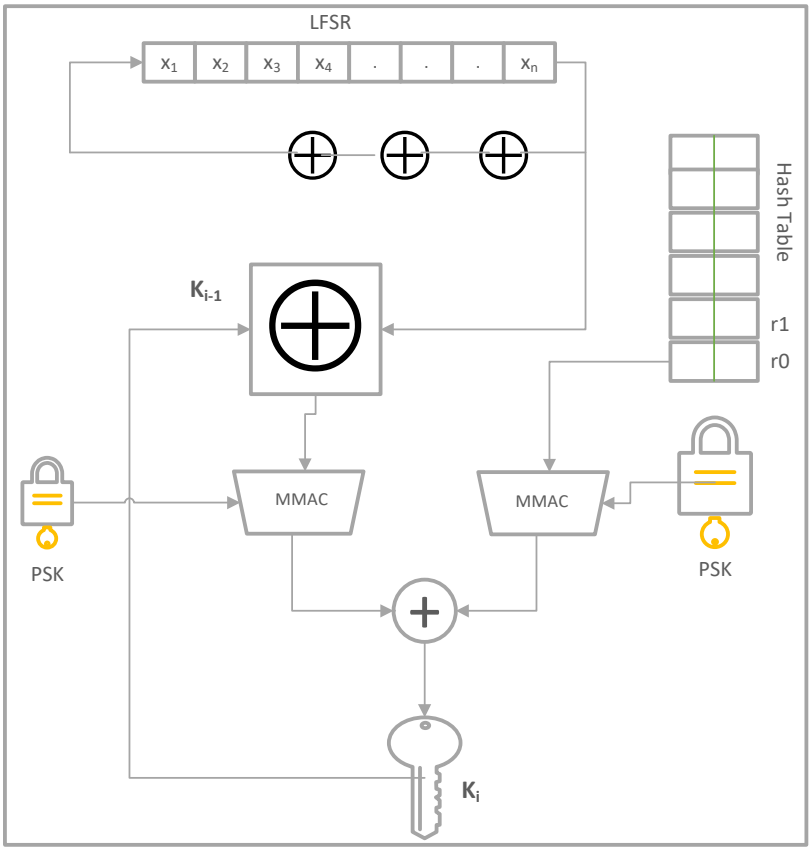


Figure 3.8: The session key generation.

Algorithm 1: Session Key Generation Algorithm.

Data: primary secret key(PSK), linear feedback shift register(LFSR), hash table(T)
Result: Dynamic Key

Calculate $K_i \leftarrow MMAC(PSK, f(x_1, x_2, x_3, ..., x_m) \oplus K_{i-1}) \oplus MMAC(PSK, r_i)$
if *all random numbers in T used up* **then**
  for i = 1 *to* n **do**
    T[i – 1] ← H(T[i] ⊕ T[i – 1])
  end
end

The output of LFSR was encrypted by a pre-shared key to generate one time password in [383] to monitor IoT devices in smarthome. But the random output from LFSR depends on the number of bits and it produces a limited random number. Therefore, we presented our approach by including a predefined random number and previously used key introduced in [385]. The dynamic session key generation presented is a lightweight process as it involves only LFSR and two MMAC operations. The MAC operation is faster and requires less processing power and memory than AES encryption[375].

3.3.2.4 Secure Communication Protocol

Generation of a symmetric key during a session ensures higher security for BSN device but a fresh key for every session is computationally expensive for IoT devices. BSN device and SDP device normally assign Key Validation Time(KVT) to a new symmetric key of a medical sensor that continuously streams physiological data such as ECG. The communication protocol is depicted in Figure 7.5 where the source device checks the KVT of the old symmetric key when it needs to transfer data. If the KVT is already expired, the source and destination device execute the session key generation algorithm and execute the authentication process using the new symmetric key. Otherwise, the source device executes the authentication process using the old symmetric key. If the authentication is successful, the source device prepares the data packet as shown in Table 3.2 and sends the packet to destination device. The source and destination device use H() of the nonce exchanged during authentication as their identification so that attackers cannot correlate session data to a BSN or SDP device.

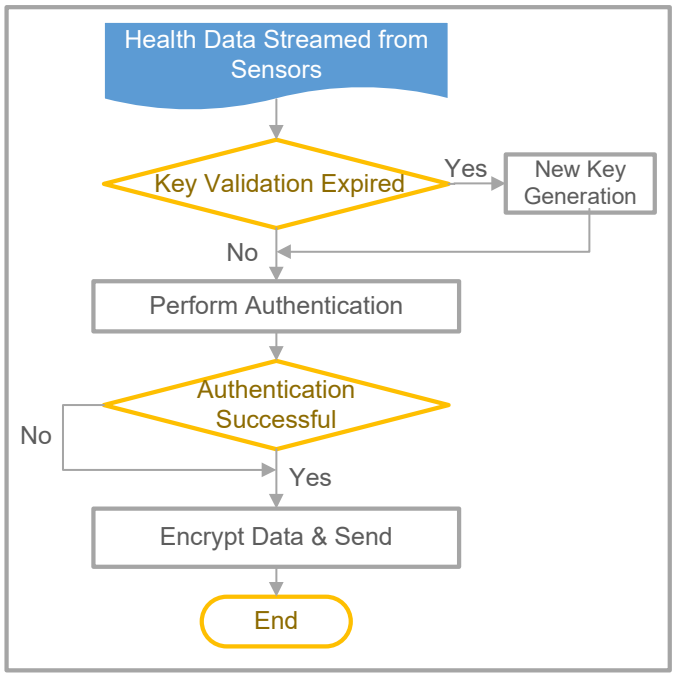


Figure 3.9: The communication protocol.

Table 3.2: The data packet format

H(Source Nonce)	H(Destination Nonce)
Sequence Number	HMAC(K_i,H(Data|| Sequence Number))
Enc(K,Data)	

3.3.3 Patient Centric Agent to Customized Blockchain

The PCA connects the patient's BSN with the customized Blockchain network. The PCA decides what data is to put into the Blockchain and, which Miner is to be selected. Blockchain is not only a tamper proof distributed database for patient's record but also an authentic platform verified by all the nodes in the Blockchain. Nodes of Blockchain might be provided by healthcare providers, other organizations or individuals. The nodes in our customized Blockchain are normally classified as half nodes, general nodes, benign nodes and miner nodes like Bitcoin Blockchain. Half nodes normally indicate an individual user or a healthcare provider that would like to use data stored in the Blockchain. General nodes are responsible for storing the chain of blocks and broadcast blocks for validation. The Miners that are powerful nodes in terms of CPU processing mine a block. Miner and general nodes ensure that a data packet originates from a legitimate node using the verification process. In our RPM architecture, benign nodes can be distinguished from other nodes by the Trust Management of Healthcare Control Unit(HCU).

The Blockchain in Bitcoin demands a lot of processing power to mine a block. Further, the transaction processing time of the Bitcoin is longer to handle stream data in continuous RPM in real time. Healthcare professionals normally need to quickly retrieve streamed data from the Blockchain. These challenges motivated us to design a customized Blockchain with Patient Centric Agent to process the patient's stream data in real time. In the customized Blockchain, the patient has full control of his or her record. The following sections contain the basic components of Blockchain technology used in the proposed architecture. The components include miner selection for the proof of work, transaction and bock. We first describe the Bitcoin Protocol before illustrating our customized version.

- Half nodes or general nodes make a transaction with the sender's signature and broadcast the transaction throughout the Blockchain network.
- Miner nodes gather certain amount of transactions and process transactions in a block. All miner nodes start solving a difficult hashing problem called Proof of Work[386] by incrementing a variable field called the nonce of the block. The Miner that successfully generates the target hash containing pre-specified number of leading zeroes first broadcasts the block to Bitcoin network and receives a financial reward for doing so.
- All nodes in Blockchain verify the block and add the block to the current Blockchain.

3.3.3.1 Miner Selection in Customized Blockchain

The Proof of Work in digital cryptocurrencies consumes huge processing power because all of the miners compete to be first to generate the target hash of block to prevent the tampering of the record. Proof of Stake and Proof of Capacity or space are alternative consensus protocols used in some cryptocurrencies.

The Proof of Stake[387] does not depend on the processing power of the miners. The Miner that owns and locks the highest share of coin to the system has the higher chance to mine the next block. For example, if there are three miners namely m_1, m_2, and m_3 which own 25%, 10% and 15% share respectively, then, the first miner builds the next block.

With the Proof of Capacity or space[388] approach, the Miner with the greatest memory or disk space capacity is selected to add the block to the chain.

In our End-to-End healthcare architecture, we propose to select a group of trusted miners based on some characteristics and the miner selection is done by the PCA. The PCA collects resource information and ratings given by other PCAs about miner nodes from TM(Trust Management) module as shown in Figure 7.4 in HCU and also send it's rating about the selected miner to TM. The PCA aggregates patient physiological data and builds a block, the block is transferred to a miner node listed in the group. The selected miner node runs Proof of Work as in Bitcoin. The process reduces the power consumption of Blockchain as one Miner produces the Target Hash.

The HCU explained in 3.3.5 executes Miner Selection Algorithm for healthcare professional's registration transaction. The traditional bank acts as a Miner on behalf of its customer for payment transactions that is discussed in Section 3.3.3.4. The Miner Selection Algorithm executed by the PCA is described in 3.3.3.1.1.

3.3.3.1.1 Characteristics Based Miner Selection We present our characteristics based Miner selection in Algorithm 2, the nonce generation for block's target hash in Algorithm 5 and block verification Algorithm in 4 respectively.

- In the proposed Miner Selection Algorithm(MSA), the PCA first obtains Miners' disk space, locked currency amount from it's NM(Network Manager) and the trust's estimation(T) from the TM of HCU. The trust's estimation(T) is discussed in 3.3.3.1.2.

- Secondly, the ratio(in percentage) of locked currency, and memory capacity of available Miners is calculated respectively.

- Thirdly, a linear equation involving locked currency and memory ratio is solved to maximize the total ratio of a miner by using linear programming subject to total ratio of the memory and currency is equal to 1 and individual share of memory or currency is equal or less than $\frac{3}{4}$.

- Fourthly, the result of the linear equation is normalized and is added with normalized trust's estimation(T) of a miner to measure the ultimate rating for the miner. Later, a set of the fittest miners are selected randomly or using hiring selection algorithm($\frac{1}{e}$ algorithm) after estimating the rating of all available miners.

- The PCA executes the algorithm to select only one miner from the set of the fittest Miners every time it has a block. In the RPM e-healthcare framework, data transaction processing rate is higher than any other Blockchain applications because of huge stream of real time data from BSN. This selection of a Miner from the list reduces the computational cost in Blockchain as well as the PCA.

- The PCA waits for a certain time after handling over the block to the selected Miner . If the PCA does not receive the block to verify as one of the validator within a pre-specified time,the PCA nominates another Miner from the fittest list. Here, the nominated Miner transmits the block to its neighbor nodes in the Blockchain network. The neighbor nodes continue

to broadcast the block in the Blockchain network. In the meantime, the Miner comes up with target hash of the block and later just broadcasts the identifier, target hash and nonce of the block in the Blockchain network. The nodes in Blockchain already having the block verify the target hash and confirm the addition of the block to the patient Blockchain. The advantage of the approach is that: patient's block can be available to healthcare professionals in real-time. The disadvantage of this is that: it causes network overhead in the Blockchain. But the availability of a patient's record in the Blockchain network is a vital requirements in RPM.

3.3.3.1.2 Trust Model The PCA needs to discover reliable nodes among the available Miners as it needs to choose only one miner to perform Proof of Work. We propose a trust model as illustrated in Figure 6.4 to discover the most reliable miners. The model executed by HCU ranks a miner on the basis of the rate given by those PCA(miner's client) that already selected the Miner, and summation of probability of some other trust parameters illustrated in Table 3.3. The PCA queries HCU to get trust's estimation (T) for a miner. The Miner's client provides the TM of HCU with feedback of the Miner regarding turn around time and mining charges information. The Miner's client that already experienced comparatively less turn around time gives a higher rating to the Miner. The Miner that voluntarily mines block is also given the highest rating.

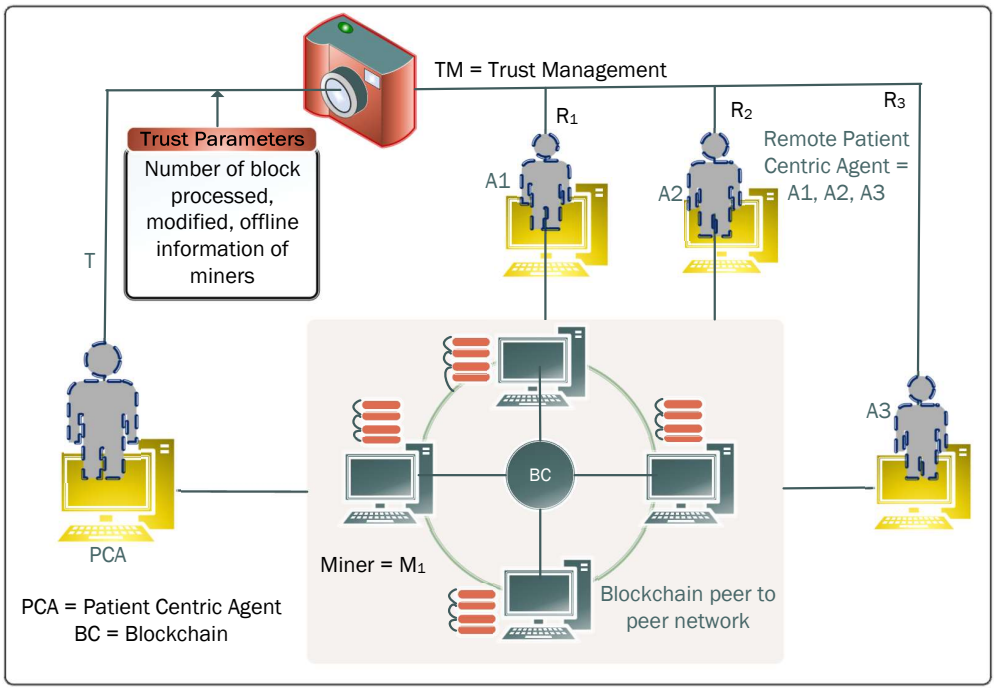


Figure 3.10: The trust model.

In Figure 6.4, let the miner m_1 is already picked up by some PCAs such as $A_1, A_2, A_3, ..., A_n$ and the normalized turn around time $TAT_1, TAT_2, TAT_3, ..., TAT_n$. The trust model is defined as in equation (3.1)

$$T_1 = d + \frac{TV_1 \times R(A_1) + TV_1 \times R(A_2) + ... + TV_1 \times R(A_n)}{(1 - d)} \tag{3.1}$$

Algorithm 2: Characteristics Based Miner Selection Algorithm.

Data: currentCurrency, currrentCapacity, turst(T[]) of m numbers of available miners
Result: List of reliable Miners

Intialize count ← 0
if *there is no available miner* **then**
    minerSelection[count + +] ← patientAgent
else
    for *each miner* i = 1 *to* m **do**
        if currentCapacity[i] ≥ Th **then**
            $\text{capacity[i]} \Leftarrow \frac{\text{currentCapacity[i]}}{\sum_{j=1}^{m} \text{currentCapacity[j]}} \times 100$
            $\text{currency[i]} \leftarrow \frac{\text{currentCurrency[i]}}{\sum_{j=1}^{m} \text{currentCurrency[j]}} \times 100$
        end
        Maximize ratings[i] ← $\text{xcapacity}_i + \text{ycurrency}_i$ subject to $x \leq \frac{4}{3}$, $y \leq \frac{4}{3}$ and
        $x + y = 1$
    end
    /* Normalize ratings and trust where A and B, C and D are the lowest and highest value of ratings[i] and T[i] respectively. 1 – R represents the scale of the normalization. */
    for *for each miner* i = 0 *to* m **do**
        $\text{ratings[i]} \leftarrow 1 + \frac{(\text{ratings[i]}-A)\times(R-1)}{(B-A)}$ $\text{T[i]} \leftarrow 1 + \frac{(\text{T[i]}-C)\times(R-1)}{(D-C)}$ ratings[i] ← ratings[i] + T[i]
    end
    numSkip ← $m \times \frac{1}{e}$
    for t = 1 *to* numTrials **do**
        Shuffle(ratings)
        bestRating ← ratings[1]
        candidate ← –1
        readyToHire ← false
        for *each miner* i = 0 *to* m **do**
            if i ≥ numSkip **then**
                readyToHire ← true
            end
            if ratings[i] > bestRating **then**
                candidate ← i
                bestRating ← ratings[i]
                if *readyToHire=true* **then**
                    break
                end
            end
        end
        if *candidate=-1* **then**
            continue;
        end
        if ratings[candidate] ≥ thresholdRating **then**
            minerSelection[count + +] ← candidate
        end
    end
end

Table 3.3: The trust model parameters

Symbol	Description
N_B	Total Number of Block within a Time Limit
N_b^p	Total Number of Block Processed by a Miner
N_b^m	Total Number of Block Modified by a Miner
$T_offline$	Total Offline Duration with 24 Hours
N_b^v	Total Number of Verified Block by a Miner
R_{TAT}	Rate from Turn Around Time
R_m	Mining Charge Rate

Where TV_1 is estimated by the summation of probability of the some parameters stated in the Table 3.3 as follows:

$TV = (1 - P(\frac{N_b^m}{N_b^p})) + (1 - P(\frac{T_offline}{24})) + P(\frac{N_b^v}{N_B})$

and $R(A_i) = WR_i + (1 - W)R_0$ where $R(A_i)$ is the rating of i^{th} client PCA.

Here, R_i is the average rate given by i^{th} client PCA on two parameters(Turn Around Time and Mining Charge Rate) and $R_i = \frac{R_i^{TAT} + R_i^m}{2}$. The client PCA defines rating R_i^{TAT}, R_i^m from 1 to 5 according to Turn Around Time and Mining Charge Rate as shown in the Table 3.4.

Table 3.4: The ratings given by individual neighbor agent

Mining Charge Rating	
Criteria	Ratings
Volunteer Mining	5
Low Mining Charge	4
Low Medium Mining Charge	3
Medium Mining Charge	2
High Mining Charge	1
Turn Around Time	
Criteria	Ratings
t_1 to t_2	5
t_3 to t_4	4
t_5 to t_6	3
t_7 to t_8	2
t_9 to t_{10}	1

R_0 is the average of previously obtained ratings from other client PCA and W is a weight in between $0 < W \leq 1$. If the current average is greater than the prior average then TM randomly assigns W: $\frac{3}{4} \leq W \leq 1$ and d is a probability factor and the value of d is $\frac{1}{N}$ where N is the number of available miners.

3.3.3.1.3 Random Miner Selection The PCA might avoid computational overhead of the characteristic based Miner selection process using Random Miner Selection(RMS) policy. However, this introduces the risk that a malicious node may be nominated. The responsibility of mining a Block given to K number of Miners can make the system secure. The PCA can let a small set of Miners to compete to mine a Block unlike Bitcoin where all Miners compete. The PCA might execute RMS in case the information from HCU for characteristic based Miner Selection is not available.

Algorithm 3: Random Selection of Miner Node.

```
   Data: list of available miner node
   Result: list of selected miner
 1 for each miner i = 1 to k do
 2 |   selectedMiner[i] ← minerList[i]
 3 end
 4 srand(time(NULL))
 5 for i = k to m do
 6 |   j ← rand()mod(i + 1)
 7 |   if j < k then
 8 |   |   selectedMiner[j] ← minerList[i]
 9 |   end
10 end
```

3.3.3.1.4 Verification of Block The Miner selected by PCA produces the target hash of block according to Algorithm 4. Target Hash is produced from Version (V), Type(T), Previous Block Hash(PBH), Timestamp(TS), Trie Tree Root(TTR), Target Difficulty(DT), Block Owner Address(BOA), and Transaction Time Frame(TTF) of the block. Next, the miner broadcasts the block for all other nodes in Blockchain network to verify the block according to Algorithm 5. Information of the next block verified by the nominated miner includes the previous block hash, difficulty level of target hash, legitimacy of all transactions and sender's payment. A Blockchain node adds the next block to the existing chain of Blocks if the verification process is successful.

3.3.3.2 Description of Transactions

In Bitcoin, a transaction is made when a sender wants to transfer cryptocurrencies to a recipient. Every transaction has two parts called input and output. Public/Private key pairs are used to hide the identity of the transaction owner. The sender and receiver are identified with their public key. A valid transaction has an authorized sender signature and a valid source of digital currency. The transaction format of Bitcoin is illustrated in Table 3.5.

Likewise, we introduce different transactions called Data Transaction(DT) for physiological data, Registration Transaction(RT) that authorizes healthcare provider such as physician, and diagnostic center, Access Grant Transaction(AGT) for granting a healthcare provider's Role Based Access(RBA) to patient's record, and Payment Transaction(PT). The health providers send the PCA patient's other records such as prescriptions, and medical test results. Healthcare provider uses the PCA's public key for preserving the confidentiality of patient's record. The PCA has authorization only to make transactions of Blockchain for patient's records. Each type of transaction is discussed in this section.

Algorithm 4: Nonce Generation Algorithm.

Data: Previous Block Hash, Difficulty Level(number of leading zero(n))
Result: Target nonce and Target tHash

```
Initialize nonce ← 0 and target ← false
Build Trie Tree of the Transactions
Run Transaction Fee Protocol
blockHeader← Hash(V||T||PBH||TS||TTR||TD||BOA||TTF)
while target = false do
    if Hash(blockHeader||nonce) = hash with leading n number zeroes then
        target ← true
        tHash ← Hash(blockHeader||nonce)
    else
        nonce + +
    end
end
return nonce|| tHash
```

Algorithm 5: Block Verification Algorithm

Data: nonce, tHash
Result: blockAcceptance

```
Initialize sigStatus ← false, dStatus ← false, iStatus ← false, tStatus ← false
dLevel← extractDifficulty(blockHeader(Difficulty))
tStaus ← checkTime(Timestamp)
sigStatus ← blockSignatureVerification()
dSatus ← checkDifficulty(dLevel)
iStatus← checkTransactionIntegrity(TrieTreeRoot)
blockHash ← Hash (V||T||PBH||TS||TTR||TD||BOA||TTF||nonce)
if sigStatus = true ∧ blockHash = tHash ∧ dStatus = true ∧ tStatus = true ∧ iStatus = true then
    blockAcceptance ← true
else
    blockAcceptance ← false
end
```

Table 3.5: The general format of Bitcoin transaction

Transaction Identifier			
Input		Output	
Bitcoin Source		Receiver Address	
Signature	Sender pubKey	Script	Receiver pubKey

3.3.3.2.1 Data Transaction **Data Transaction(DT)** format is illustrated in Table 3.6. SDP device(smartphone) makes DT that consists of physiological data coming from BSN during a time interval (t_i to t_j).

The SDP device puts it signature in DT's Signature field and sends DT to Patient Centric Agent(PCA). The PCA signs the DT after the verification of SDP device's signature. The signature verification process is illustrated in Fig. 6.11

Table 3.6: The format of Data Transaction

TimeStamp	SensorId
SDP Address	PCA Address
MultiSignature	
Record Type	
Hash of Cipher Data	
Cipher of Data	
Transaction Fee	
CREDITLIMIT	CREDIT
CREDITPRICE	

The multi-signature script introduced in[389] for the field **MultiSignature** in DT is defined as follows:

rScript = OP_1pubKey$_m$||pubKey$_a$||OP_2||OP_CHECKMULTISIG

Where, OP_1(n) indicates the required number of signatures among OP_2(m) numbers of signatures. pubKey$_m$ and pubKey$_a$ represent the public key of SDP device and PCA respectively.

The signature formation is illustrated in Figure 6.3. A SDP device generates hash from the header of a transaction and then encrypts that hash using its private key. The signature built by a SDP device is as follows: MultiSignature$_m$ = Enc(privateKey$_m$, H(Timestamp||SensorId|| SDP Address||Hash of Data||Transaction Fee)).

The PCA checks the signature and again encrypts the signature with its private key to obtain MultiSignature = Enc(privateKey$_a$, MultiSignature$_m$). The node that verifies the signature first decrypts the signature using PCA's public key, next, it decrypts the hash by SDP device's public key.

The DT contains the encrypted health data called **Cipher of Data**, the hash code of the encrypted data(Hash of Cipher Data) to ensure data integrity. The **Transaction Fee** varies as every transaction doesn't carry the same amount of health data. Three fields called CREDITLIMITS, CREDIT, and CREDITRICE are used to estimate Transaction Fee. CREDITLIMITS represents the maximum amount of credits for a transaction such as Data Transaction, Registration Transaction. Further, the health data in a specific transaction such as Data Transaction might vary. So, CREDIT represent the required amount for processing of a particular transaction. CREDITPRICE represents the price per byte in a transaction. For instance, the PCA sets CREDITLIMITS(2000)and CREDITPRICE(100 Credit) for a DT with 10 bytes, the CREDIT required to process the transaction is $10 \times 100 = 200$ VCP(Virtual Credit Point).

3.3.3.2.2 Registration Transaction The Registration Transaction(RT) format is illustrated in Table 3.7. RT represents the legitimate healthcare provider. The RT is issued by Healthcare Control

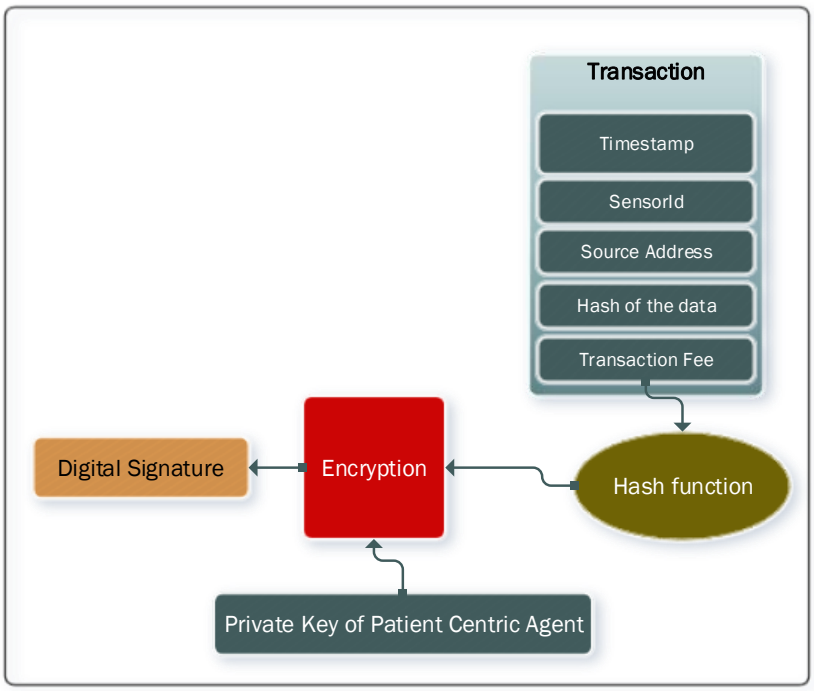


Figure 3.11: The signature formation process.

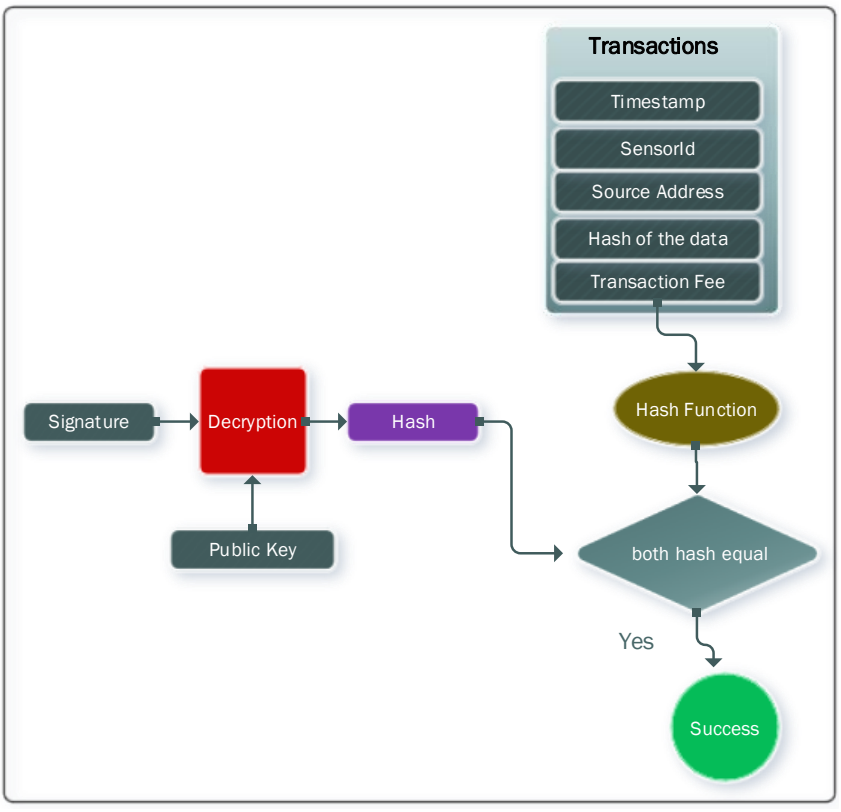


Figure 3.12: The signature verification process.

Unit(HCU) and stored in Blockchain. The signature of HCU in RT ensures the legitimacy of healthcare provider.

Table 3.7: The format of Registration Transaction

Record Type	Timestamp
HCU Address	
HCU Signature	
Healthcare Provider Signature	
Healthcare Provider Adress	
Healthcare Provider Profile	
Transaction Fee	
CREDITLIMIT	CREDIT
CREDITPRICE	

3.3.3.2.3 Access Grant Transaction The Access Grant Transaction(AGT) is shown in Table 3.8. The PCA separates Input and Output in AGT like Bitcoin transaction. The PCA includes MultiSignature of data source in Input and a Cipher for healthcare provider to access data in Output. The Cipher includes dynamically constructed Patient Record Encryption Key(PREK) described in 3.3.3.2.4, DeviceMetaData such as Medical Sensor Id, Data Window Time Frame etc., and Time that indicates the validation period of Patient Record Encryption Key. The PCA produces the Cipher using the healthcare provider's public key so that only the legitimate healthcare provider obtains the access to health data. The AGT also contains an Access Granting Code(AGC) and Record Type explained in 3.3.3.2.4. The AGC varies based on the role of healthcare provider. For instance, the AGC for nurse is different from that of a physician.

Table 3.8: The format of Grant Access Transaction

Record Type	Timestamp
Access Granting Code	
Input	Output
Source Address	Destination Address
MultiSignature	Enc($pubKey_{dest}$, PREK || deviceMetadata||Time)
Transaction Fee	
CREDITLIMIT	CREDIT
CREDITPRICE	

3.3.3.2.4 Patient Record Encryption Key Generation The PCA makes Data Transaction(DT) by gathering data in predefined time frames. The challenge is to encrypt every transaction by using

individual keys so that healthcare professionals can access only limited records that are assigned. One key assigned to a medical sensor might give healthcare provider access to huge records for a long time. Encryption of transaction according to its time frame window ensures fine granular access of patient's record. In addition, healthcare providers have different access levels based on their roles. The PCA is also required to construct Patient Record Encryption Key(PREK) based on Record Type(RT) and healthcare provider's role. The PCA needs to dynamically construct Patient Record Encryption Key(PREK) during processing a transaction as the storage of individual keys per transaction requires huge memory.

The PCA produces the PREK through Hash operation of its Secret Key, Sensor ID, and Time Frame that includes the date and window time frame($22-03-2018 : 10.30-10.45$) of a transaction.

$PREK = H(\text{PCA Secret Key}||\text{Sensor ID}||\text{Record Type}||\text{Time Frame})$

Where PREK represents Patient Record Encryption Key for a pre-specified time frame. The PCA encrypts a transaction by using a dynamically constructed PREK. PREK can be regenerated by the PCA whenever the PCA grants a healthcare provider access to patient health records. The PCA can share its secret key with SDP and BSN so that BSN and SDP can encrypt physiological data generating PREK. As PREK involves only H() operation, generation of PREK is also feasible for BSN.

The Healthcare Provider's Wallet can only decrypt the patient's record and the HPW deletes the record after elapse of time in AGT illustrated in Table 3.8. In RPM, healthcare professionals deal with diverse genre of patient records such as raw records, prescription records, and diagnostic result. PCA assigns a code to a patient's record: raw record(00), prescription record(01), diagnostic result(10) etc.

The PCA defines the user of health data based on the healthcare provider'roles drawn from an eHealth standard such as openEHR[390]. This includes Healthcare Provider Organization such as Diagnostic Center(DC) or Hospital(H), Individual Healthcare Provider such as Physician(P) or Nurse(N), and Healthcare Consumer such as Relatives(R) or Others(O). Finally, the PCA merges patient's record code and selected healthcare user code to make Access Granting Code(AGC) of AGT shown in Table 3.8 . For instance, if the PCA gives a patient's Physician, Nurse and Relatives access to medication prescription, the PCA produces the code(01-110010) illustrated in Table 3.9. The AGC and the HPW will reject access if someone's role does not satisfy the AGC.

Table 3.9: The Access Granting Code

RT	P	N	DC	H	R	O
01	1	1	0	0	1	0

3.3.3.3 Data Block Structure

Bitcoin Miners collect transactions from different users worldwide and build a block with 1024 transactions. The Miner creates a Merkle Tree that is a binary tree to pack the transactions in a block. The Merkle Tree ensures the integrity of the transactions in a block. A Blockchain with Merkle Tree in Bitcoin is depicted in Figure 6.12. Here, we assume that there are four transactions: $Tx1, ..., tx4$. To create Merkle Tree, first, a hash value for each transaction is produced. Secondly, hash function is applied on the concatenated hash value of two transactions and this process is continued until only one hash value is generated from all the transactions. The final hash value is

the root of the Merkle tree. The Merkle Tree root is inserted into one field of the block. The nonce field that is also called counter is the only variable in the block. Nonce is incremented by the miner as one of the inputs of hash function until it produces a target hash of the block. The previous hash field contains the target hash value of the latest block of the blockchain. In this way, a chain of blocks is created, which protects an individual block to be tampered.

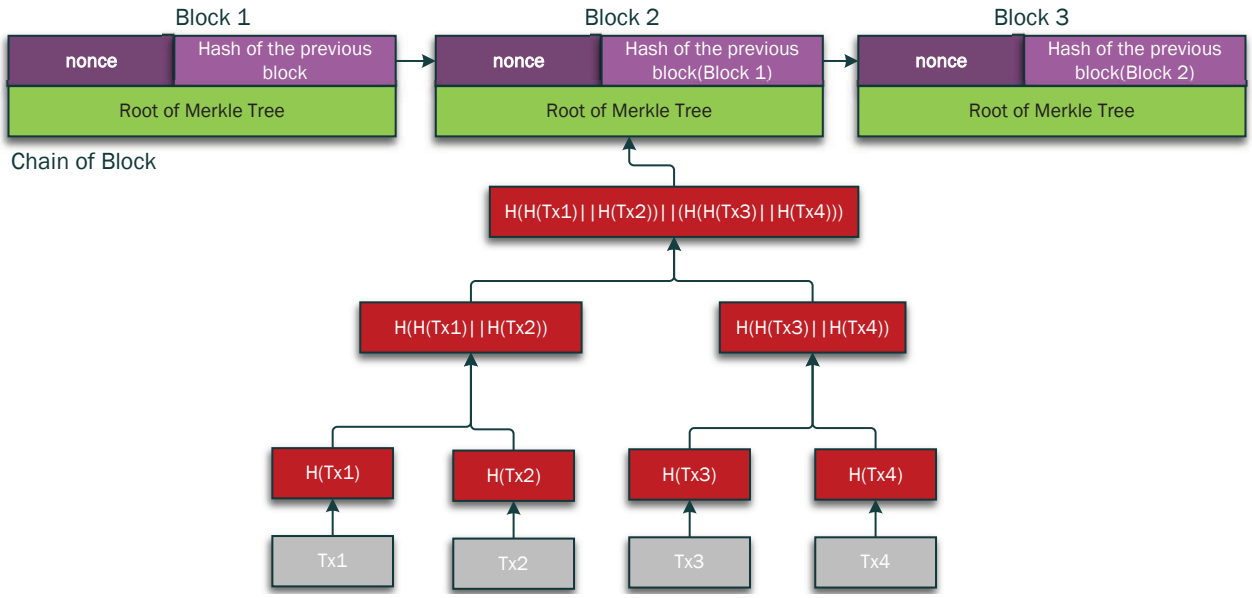


Figure 3.13: The Blockchain in Bitcoin

In our customized Blockchain for IoT healthcare framework, we propose a Trie Tree instead of a Merkle Tree to retrieve data quickly while maintaining the integrity of data. Data Transaction Block(DTB) consists of only physiological data transactions. Similarly, other Blocks such as Registration Transactions(RT), Access Grant Transactions(AGT) consist of respective kinds of transactions. Fields of a Data Block are depicted in Table 6.6. The Type in Block represents the kind of transactions (DT, RT, AGT). The PCA inserts a Trie Tree root in place of the Merkle Tree root in the Block. The Merkle Tree demands huge processing power and is not suitable for RPM because of huge volume of streamed data and takes longer time to retrieve data to preserve data integrity.

The transaction in Block is arranged in the Trie Tree according to a device identifier. The leaf of Trie Tree holds the hash value of the transactions in the Block. In Figure 6.10, we show that there are three sensor devices namely EEG,EMG, and HRV and every alphanumeric character of a sensor identifier creates a label in the tree. Transactions of a medical sensor is labeled as T1 – T2, T2 – T3,...,Tn – Tn + 1 at leaves of that sensor according to transaction generated time frame window. In Figure 6.10, the parent node of the leaves contain the hash value(H(H(TX1)||H(TX2)||H(TX3)||H(TX4)...)) of the concatenation of its children hash value. Likewise, an ancestor node contains the hash value of the concatenation of its descendants' hash value as well as it's label. The significant advantage of Trie Tree lies is that ; transactions can be searched at the complexity that is equal to the length(L) of a sensor identification O(|L|). Trie Tree also preserves the integrity of data as parent node and leaf node contain hash value of the transaction. We can check the integrity of the transactions just by observing the root like Merkle tree. Further, Trie Tree involves fewer hash operations than Merkle Tree.

Table 3.10: The format of Data Block

Block Header of Blockchain	
Field	Description
Version	Block Version Number
Type	Transaction type
Previous Block Hash	Hash of the previous block in the chain
Timestamp	Creation time of the block
Trie Tree Root	Root of the Trie tree containing transaction
Target Difficulty	The Proof-of-Work difficulty target
Nonce	A counter for the Proof-of-Work
Block Owner Address	
Transactions Timeframe	

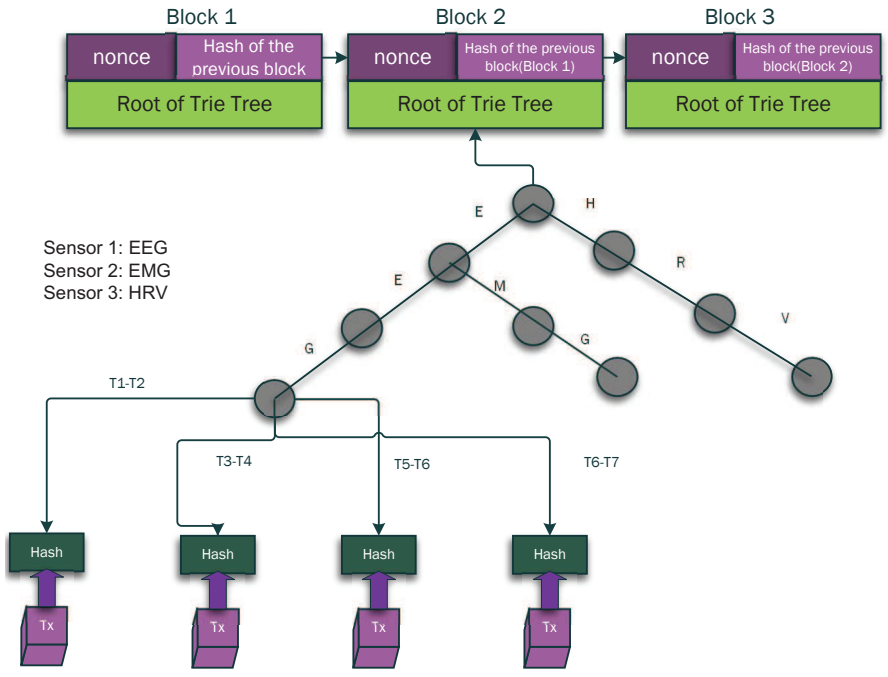


Figure 3.14: Trie tree structure

3.3.3.4 Transaction Fee Protocol:

In IoT Blockchain healthcare, the management of huge streamed data is prime target to make the system efficient and effective. Blockchain in IoT healthcare also need to deal with Transaction Fee and healthcare provider's fee in a secure manner. Digital Currency(DC) in Bitcoin or Ethereum is still not as widespread as traditional currency. Therefore, we propose to incorporate the conventional banking system into our IoT Blockchain healthcare system. In the proposed payment protocol, we assume that the patient, Miner and healthcare provider own Virtual Credit Account(VCA) in the Banking system. Every node in Blockchain network is associated with one or more traditional banks. Due to security threat, traditional Smart Card/Credit Card for financial transactions are discarded in favour of virtual credit. The PCA buys Virtual Credit Pints(VCP) from a bank using the patient's Smart Card/ Credit Card. The payment protocol of the proposed e-healthcare system is illustrated in Figure 6.6. and is described as follows:

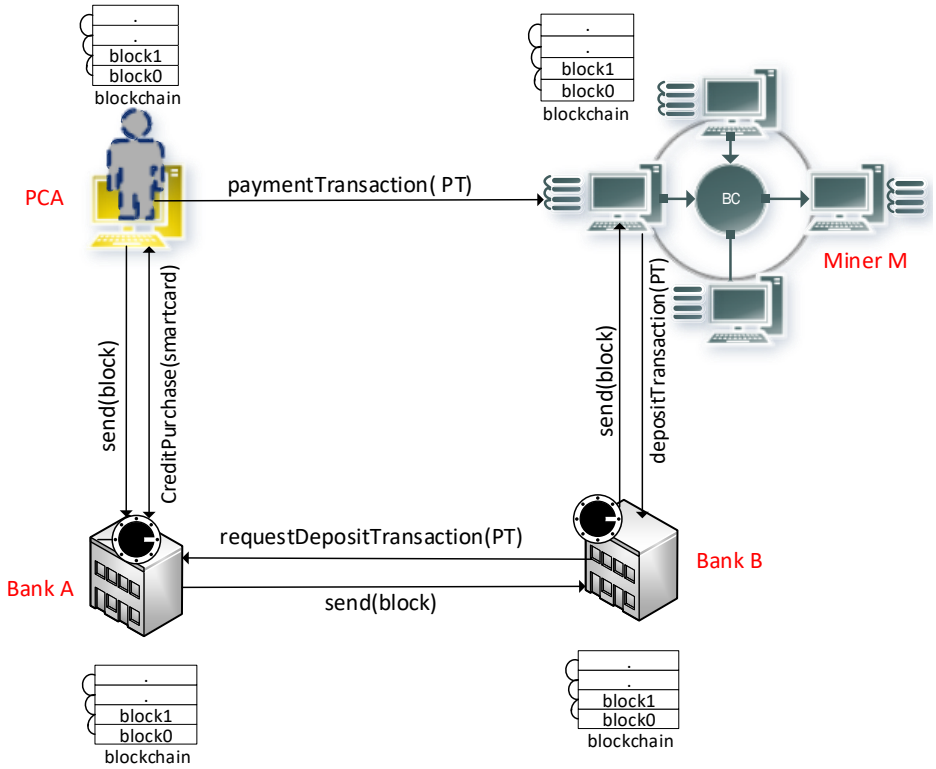


Figure 3.15: Payment protocol

1. Patient Centric Agent(PCA) purchases Credit Points(CP) from Bank A in exchange for traditional currency.

2. PCA constructs a Payment Transaction(PT) as shown in Figure 6.5 . The PT holds PCA's signature and Bank A's signature. The PCA sends PT to Miner M nominated by the PCA for target hash generation.

3. The Miner M puts its signature after verification of the PCA's signature and Bank A's signature in PT. The Miner M transfers the PT(Payment Transaction) to Miner's Bank B.

4. Bank B verifies all signatures on the PT and inserts its signature into the PT. The Bank B requests Bank A to make a Deposit Transaction(DT) for Miner M.

5. Bank A prepares a new transaction called UTXO(Unspent transaction) for the PCA and Output Transaction(OT) for Miner M. Finally, Bank A builds a block with all transactions produced to complete the payment and sends the block leaving the Previous Transaction Hash field empty to Miner M and the PCA.

6. The PCA and Miner M generate their respective hash value by placing the Previous Hash Block in local Blockchain. Banks in Blockchain maintain a global Blockchain for processing of UTXO and OT like Bitcoin.

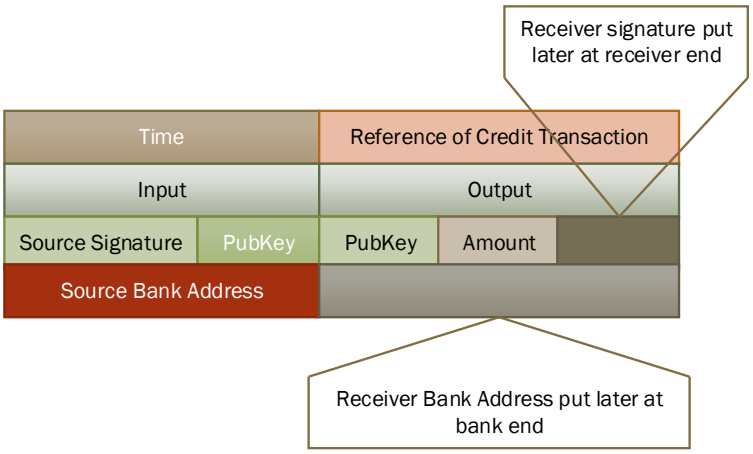


Figure 3.16: Format of Payment Transaction

3.3.4 Customized Blockchain to Healthcare Provider

3.3.4.1 Healthcare Provider Agent(HPA):

Healthcare Provider Agent is healthcare provider Centric server to store and analyze patient health data. The HPA's functionalities are similar to PCA. For example, HPA nominates a Miner in Blockchain and performs Key Management at the healthcare's end etc.

3.3.4.2 Healthcare Provider Wallet(HPW):

The HPW has three modules called Blockchain Interface Module(BIM), Registration Interface Module(RIM) and Ex-filtration Detection Module(EDM) shown in Figure 7.4. BIM(Blockchain Interface Module) provides the healthcare provider with Blockchain access and processes transactions for the Blockchain. EDM(Ex-filtration Detection Module) proposed in [391], [392] prevents insider attacker from breaching patient's information. Patient health data privacy can be breached by the healthcare service providers in attacks known as Insider Attacks. The RIM(Registration Interface Module) performs healthcare provider registration with the PCA(Patient Centric Agent) and HCU(Healthcare Control Unit) to safeguard against healthcare provider attempts to use patient's data without permission.

3.3.5 Healthcare Control Unit

Healthcare Control Unit placed in the upper tier of the RPM architecture proposed here is a Trust Center for heathcare provider and the PCA. Trust Management(TM) of HCU monitors the activities of miners and the PCA in Blockchain, authorizes and certifies healthcare providers.

3.4 Performance Analysis

In this section, firstly, we analysis the performance of proposed Patient Centric Agent based monitoring architecture in terms of energy and End to End delay. Secondly, we discuss the security strength of the architecture in terms of some common attacks. After that, the simulation environment and results for the performance analysis are presented.

3.4.1 End to End Energy Analysis

In our architecture, we allocate less processing to the body area sensor devices because of energy and processing power constraints. The sensor devices in BSN generate symmetric key in lightweight computation for authentication. The authentication process involves HMAC operation and voice identification module which are not expensive in terms of speed and energy consumption. In the architecture, SDP device only receives physiological data from BSN and send data to the PCA . As SDP device such as smartphone is also energy constrained and memory limited, we let SDP device run only cryptography related algorithm to transfer data securely to PCA. Classification of physiological data, Block Generation, Block Verification, communication with Blockchain are some energy and processing power-hungry tasks and those are accomplished by Patient Centric Agent. This ensures more reliable processing of patient health data than traditional architecture where mobile devices normally act as coordinating node and the device has high chance to fail. Data that the PCA deems uneventful will not be stored in the Blockchain resulting in the consumption of less energy than conventional Blockchain architectures. Since, the PCA nominates only one Miner node to mine a block, the overall energy consumption of the customized Blockchain is further reduced. In addition, we propose Trie Tree based transaction packing where fewer hashing operations are required leading to further energy savings.

3.4.2 End to End Delay

Bitcoin processes around 3to4 and Ethereum[74] processes around 20 transactions per second. Normally, the number of processing transactions per second depends on the consensus process and difficulty level of Target Hash and Hash(Ethash, SHA-3, Blake2 etc.). In our customized Blockchain, the PCA selects only one miner. Hence, Blockchain does not demand the high difficulty level because of the absence of minor competition. So, the number of transactions per second in our Blockchain is more than that in Bitcoin. In emergency cases, the PCA can bypass the Blockchain and directly send data to an authorized healthcare provider. Later, the PCA can store the emergency data into the Blockchain. This approach also significantly improves the End to End delay of health data processing. In addition, a patient record can be quickly retrieved from the Trie Tree which also helps the minimum End to End response.

3.4.3 Attack Analysis

1. **Man in the Middle attack:** Man in the middle attack normally happens when sender and receiver exchange keys. In our architecture, we let devices at different segment come up with the same key during every session to safeguard against man in the attack.

2. **Replay attack:** HMAC authentication is susceptible to a replay attack if it is not modified by some other means. An authentication protocol with time and session random number is designed to prevent an attacker from replaying.

3. **Eavesdropping:** The channel between BSN devices, SDP devices and PCA exchanges encrypted health data. So, attackers cannot modify health data after intercepting data packet. Attackers cannot gain knowledge about the source and destination from the intercepted data packet because of dynamic identification.

4. **Spoofing attack:** Attackers sometimes change the identity of the data owner; this is known as a spoofing attack. In our architecture, the source and destination agree on dynamic identification and GPS while performing authentication . As a result, an attacker cannot inject the wrong source address or destination address. The Mining fee discourages attackers from making a fake transaction however the attempt would be discarded anyway during the verification stages owing to invalid signature.

5. **Compromised Key attack:** As described above, periodically generated symmetric key is used to perform authentication among devices at different segment. Attackers cannot have the key without capturing hardware and software control of the devices. BSN allows access of the device based on proximity. Therefore, attacker will not able to get unauthorized access to BSN device because of its physical location. Attacker can control all devices by compromising one device if only one shared key is used by all BSN and SDP devices. So, we consider device wise dynamic key generation. In this case, even adversary compromises one device, other devices are still protective from the attack. Moreover, the Security Service Module of PCA analyzes the network traffic from SDP and BSN to separate the affected device at patient's end.

6. **Denial of Service attack:** A DoS attack cannot succeed in Blockchain because attackers cannot stop the activities of all the nodes in the Blockchain network by sending fake blocks. BSN and SDP are safe from DoS attack because the PCA blocks fake requests and all traffic goes through the PCA. In Blockchain, although the PCA and SDP devices are susceptible to denial of service attack, due to patients intervention, such attack can be mitigated.

7. **Patient Privacy:** Patient Centric Agent can preserve patient's privacy using public key/private key encryption in Blockchain. Blockchain processes, verifies and stores block anonymously. In Blockchain, attackers cannot link patient's prescription record to patient's relevant physiological data. As patient's identity of real-life is hidden in the system, the attacker does not benefit even if it gains some data access. Further, BSN device, SDP device and PCA communicate with each other using their sessional identifier. Consequently, session identification helps patients conceal the device's real identification to attackers.

8. **Reliable Service:** Our system provides reliable service for the patient. The Blockchain is a distributed ledger and open to all. Consequently, an attacker might claim to be a specialist healthcare professionals to gain the patients data or to earn money. Further, patients prefer

a healthcare provider with a good reputation. So, an attacker might appear as a reputed healthcare professional. To safeguard against this, we propose Healthcare Control Unit that authorizes legitimate healthcare professionals.

3.4.4 Simulation Environment and Results

First, we discuss the simulation environment and performance of Miner Selection Algorithm executed in PCA. Later, we discuss the performance of security protocol at patient's end(BSN, SDP and PCA).

3.4.4.1 Simulation & Performance analysis for Miner Selection Algorithm in customized Blockchain

We implement the Miner Selection Algorithm executed by the PCA using Java programming environment. We use Java 8 Development Kit 64 bit and Netbeans IDE 8.1 as editor. We ran the Bitcoin proof of work on three machines specified in Table 7.1. Profiler of Netbeans IDE 8.1 act as performance analysis tools in our simulation. We analyze the performances of our Miner Selection Algorithm(MSA) with Ethash used in Ethereum [74] as Proof of Work and Bitcoin Proof of Work in terms of CPU time and memory. Ethash in Ethereum is faster than SHA-3 used in Bitcoin. Ethash is memory bound operation whereas SHA-3 is CPU bound operation[74]. In Java profile, we can monitor the CPU time and memory of the host machine consumed by the application program. We define the following metrics for the performance evaluation:

Table 3.11: The Miner specification

SL No	Component	Description
M_1	Processor	Intel(R)Core(TM)I3-2310M CPU@2.10 GHz 2.10
	Memory	4.00GB
M_2	Processor	Intel(R)Core(TM)I5-7200U CPU@2.50 GHz 2.71
	Memory	8.00GB
M_3	Processor	Intel(R)Core(TM)I7-4770 CPU@3.40 GHz 3.40
	Memory	16.00GB

CPU Time Monitoring represents the required amount of CPU time to execute a program. The dark line indicates the percentage of CPU usage of the specific application. The CPU time for individual method of an application is traced in the profiling tools.

Memory Monitoring indicates the amount of heap used by an application. The light portion estimates the available heap and the dark portion estimates the amount used by the dynamic objects.

Surviving Generations indicates the number of generations that are currently alive on the heap where generation means a set of instances produced between two garbage collections.

In our simulation, we consider three miners and a Patient Centric Agent as Clients-Server where all the clients act as miners send the necessary information to the PCA to calculate the ratings of the miner for the selection process. Blocks with different numbers of transactions(252, 512, 1024) is used in the simulation. The number of leading zeroes in the Target Hash was set at 6. The telemetry view of the MSA and PoW(Proof of Work) in our Continuous Patient Monitoring with a Patient Centric Agent(CPMwPCA) is illustrated in Figure 3.17. The proof of work of Bitcoin

is executed in three miner nodes and the telemetry view of two of those miners are illustrated in Figure 3.18 and Figure 3.19. It is observed that the memory and CPU utilization of the Bitcoin proof of work is higher than our MSA+PoW(the proposed miner selection algorithm and Proof of Work using Ethash utilized around 25% of CPU and 98MB memory whereas the average CPU utilization and memory of three miners in Bitcoin Proof of Work is around 45% and 195MB respectively). The MSA is executed for the first block. The PCA does not run MSA for the rest of the blocks. We use Trie Tree to store transactions, which incurs less cost than Merkle Tree and only one machine executes the Proof of Work. As a result, The proposed solutions significantly save power consumption of the Blockchain. Power saving is appropriate for a personalized Blockchain like remote patient monitoring where individuals, government, different institutions, and healthcare providers contribute to Blockchain's node. In Figure 3.20, we show a comparison between CPU time MSA+PoW and Bitcoin Proof of Work. Our algorithm improves over the Bitcoin PoW because we select a group of miners when MSA is executed. Later, we let them mine patient data transactions one by one and Ethash are faster than SHA-3 in Bitcoin. Further,as the system allows only one miner to generate the target hash of the block, the system does not require to increase the difficulty level with the addition of new miners. Difficulty level remains constant over time.

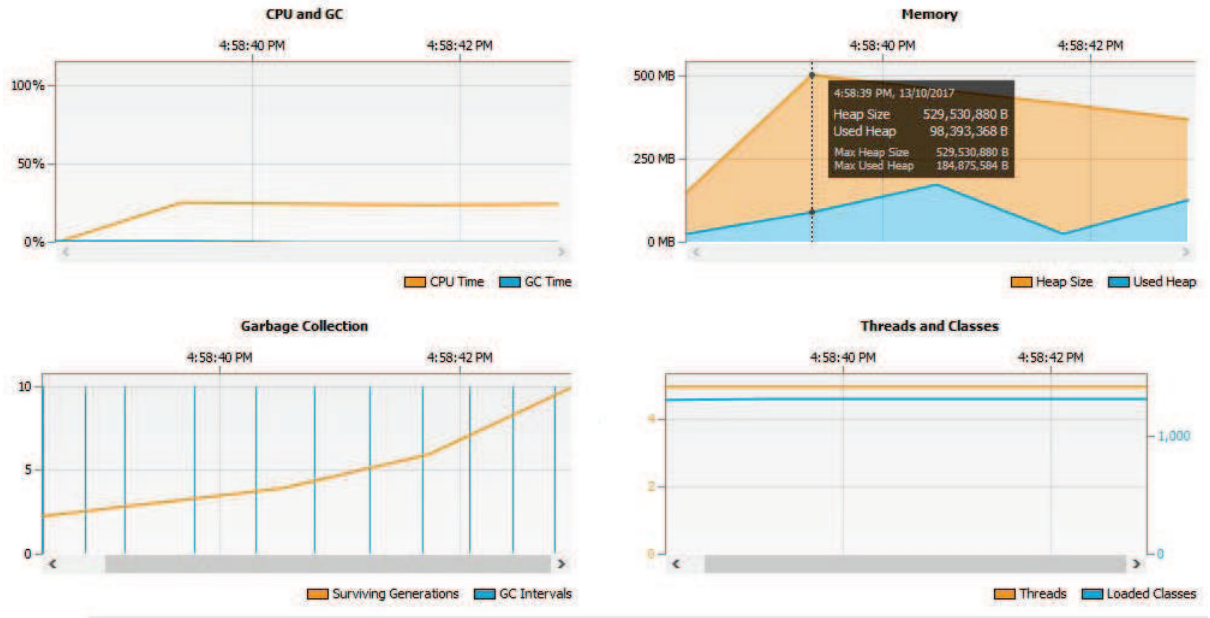


Figure 3.17: The VM Telemetry of CPMwPCA PoW in Miner 2.

3.4.4.2 Performance Analysis of Security Protocol at Patient End

Our security protocol at the patient's end is implemented in Intel(R) Core(TM) i5-6500CPU@3.20GHz machine by using Java. We show the comparative study of performance analysis of our protocol with ACLF[393] and BSN-Care[339] in terms of reliability, a number of error packets and throughput. The reliability of the proposed security protocol as depicted in Figure 3.21(a) improves over the ACLF and BSN-Care because of our lightweight proximity based authentication, and multilevel storage(Patient Local Server, Cloud, Blockchain). In ACLF, the pre-deployment of the triple key is applied for the lifetime of the sensor, therefore, if the key is exposed to attackers, adversaries might control all of the devices and hence it reduces the normal rate of transferring data

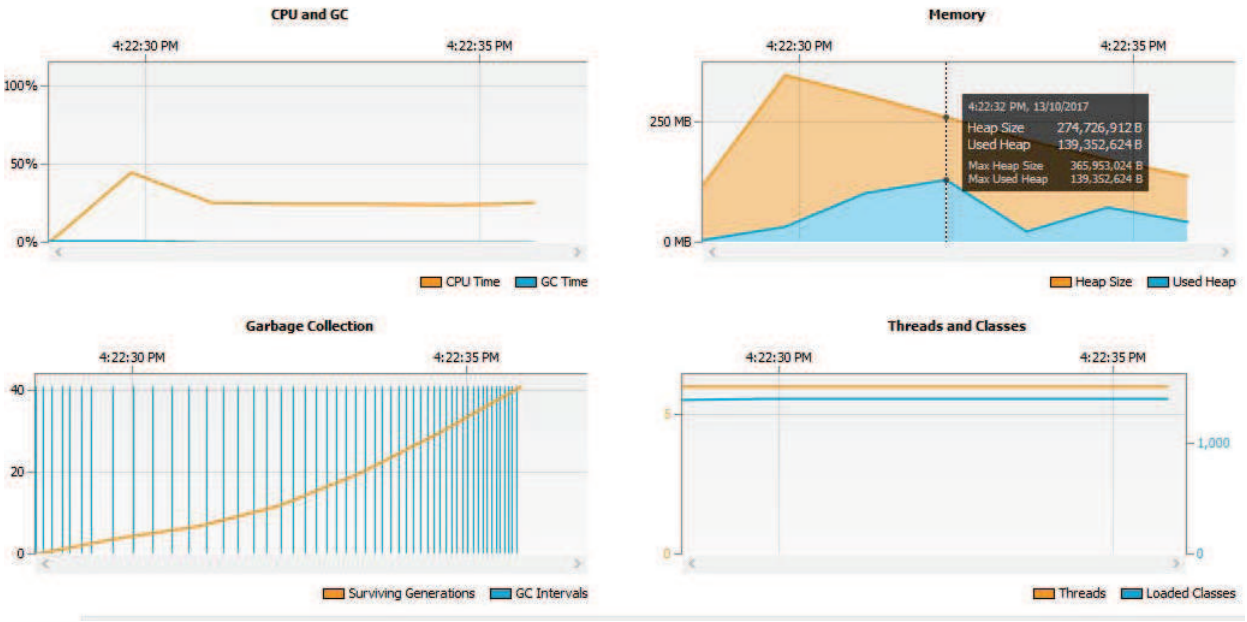


Figure 3.18: The VM Telemetry of Bitcoin PoW in Miner 2.

Figure 3.19: The VM Telemetry of Bitcoin PoW in Miner 3.

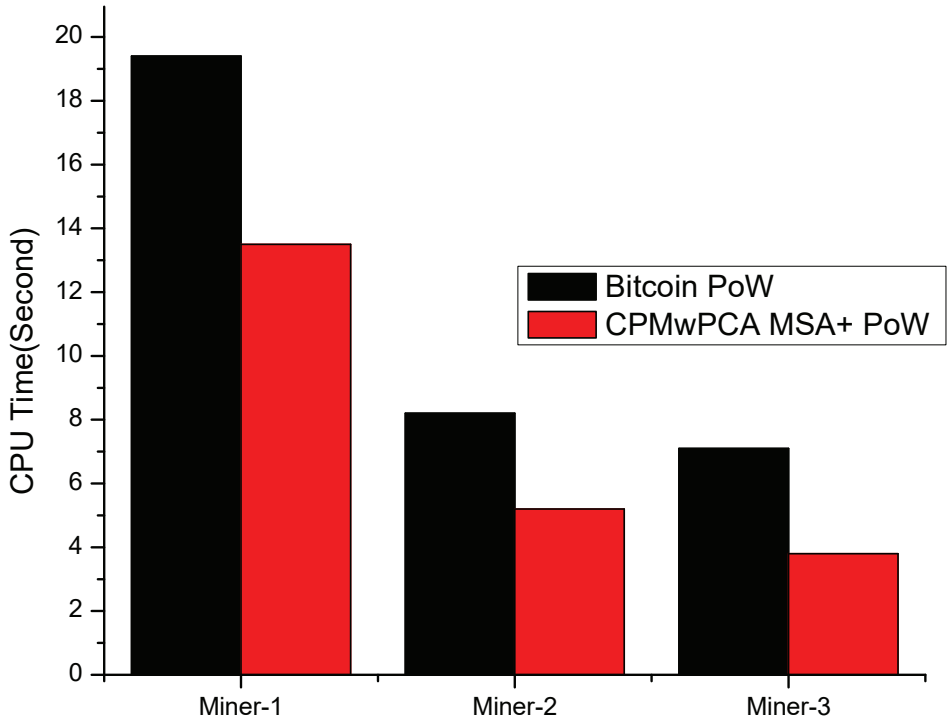
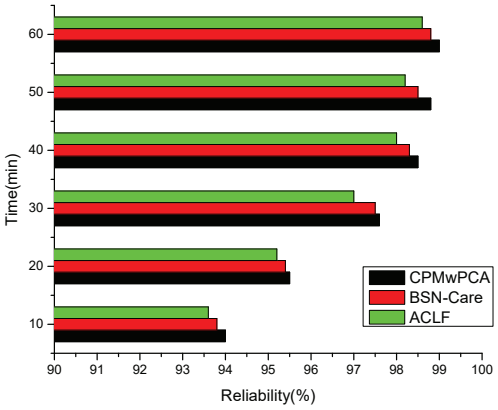
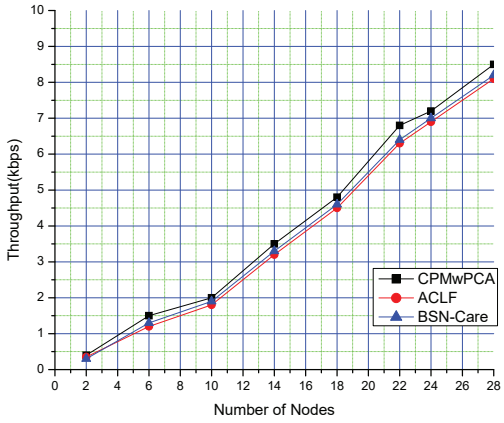


Figure 3.20: The CPU time comparison of Bitcoin PoW and CPMwPCA MSA+PoW.

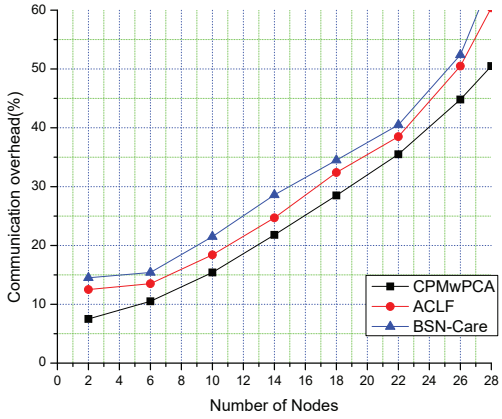
in ACLF that affects the throughput illustrated in Figure 3.21(b). In contrast, the BSN-Care proposed single server for the storage of all patients' record and therefore network congestion reduces the throughput. In CPMwCPA, we present the periodically generated key mechanism instead of sharing information except during deployment to protect the devices from eavesdropping which reduces the communication overhead as depicted in Figure 3.21(c). In CPMwPCA, proximity based authentication ensures SDP devices receive data from legitimate BSN devices and the probability of receiving error packet as shown in Figure 3.21(d) is comparatively low. On the other hand, neither ACLF nor BSN-Care consider proximity authentication in their security proposal.



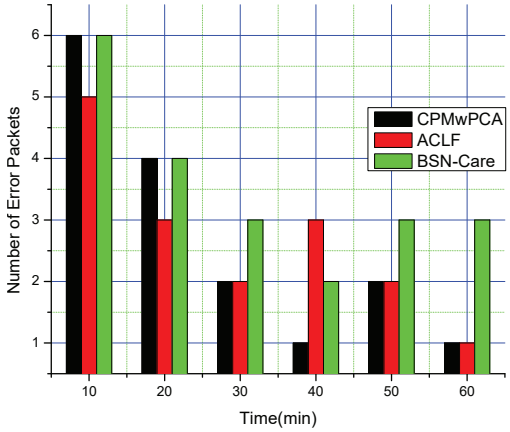
(a) The Comparison of reliability.



(b) The Comparison of throughput.



(c) The Comparison of communication overhead.



(d) The Comparison of packet error.

Figure 3.21: Communication performance analysis

3.5 Discussion on Validation of Simulated Results

In this chapter, we introduced the Patient Centric Agent (PCA), an autonomous software agent that connects body area sensor networks to Blockchain. To supervise and mediate data flow between body area sensor networks and the Blockchain, the PCA uses a secure communication and access control protocol. In addition, the PCA is in charge of managing the blockchain's consensus protocol. The core protocols of Blockchain, including consensus mechanism, block structure, and access control have been modified to make it fit for storing health related data. Since we modified several components of a Blockchain, we needed to implement our own Blockchain using Java/Python, REST API, JSON and JavaScript. We compared the core components of our customized Blockchain with the Bitcoin Blockchain because we modified the consensus mechanism and data block structure of the Bitcoin.

We analysed the result of the protocols of our approach and Bitcoin with respect to several performance parameters including CPU time monitoring and memory monitoring using NetBean's profiler. The result shows that the PCA controlled Blockchain can save power consumption and increase throughput in our simulation environment.

To analyse the performance of the communication protocol including access control, we implemented the protocol using Java Cryptography API and compared the results with other counterpart existing protocols. The results were analysed with respect to a couple of performance metrics including reliability, throughput, communication overhead and packet error. Further, we aim to incorporate the PCA with Ethereum Blockchain through smart contracts and IPFS protocol to incorporate a large amount of health data.

To validate and verify the simulated results mentioned above, we can follow the Design Science Research (DSR) methodology. This requires the development of the framework in real-life or design a testbed for verifying the simulated results. DSR [394] refers to an outcome-based research validation method which provides a concise and logical method for addressing established research questions through data collection, interpretation, evaluation, and discussion. This method examines the interplay of the framework with its other components to assess the overall system's function. This entails the gradual evolution of the framework in a real-world setting. DSR is categorized into two kinds: problem-oriented and solution oriented. Problem oriented DSR focus mostly on social science, such as research into real-world investigations. Solution-oriented DSR includes designing and validating an artifact and emphasises technical analysis.

3.6 Conclusion

In this paper, we present a Patient-Centric Agent based healthcare architecture. The architecture consists of BSN, Smartphone(Sensor Data Provider), Patient Centric Agent, Blockchain, and Healthcare Provider Interface. There are multiple communication channels from End to End of this architecture such as BSN to Smartphone, Smartphone to PCA, PCA to Blockchain. Every channel requires security against different network attacks such eavesdropping, Sybil, and man in middle. Further, BSN is a power constraint network in eHealthcare architecture. High computational encryption and authentication are not appropriate for BSN network. So, our research focuses on the proposal of lightweight encryption and authentication for BSN to Smartphone channel as well as Smartphone to PCA. Secondly, BSN produces a huge stream of data and needs to perform some pre-processing on data before sending data to Blockchain. Further, the processing rate of a block produced from real-time data might be slower than that of data arrival in Blockchain. Therefore,

we focus on the development of an intelligent Patient Centric Agent that coordinates among the BSN and Smartphone. The PCA categorizes patient's data as eventful and uneventful, defines security level, controls access for patient data and generates alarms during the emergency, nominates miners in Blockchain to optimize the overall energy of the customized Blockchain. Blockchain network confirms the privacy of patient documents, tamper-proof, availability, and guards against a single point of failure. Energy and security analysis of the proposed architecture was done to demonstrate its applicability in continuous health monitoring system.

In this work, We modified some core components of Blockchain including consensus mechanism, packing transactions in Trie tree instead of Merkle tree, access control, block data structures etc. To analyse the performances of these algorithms, we need to simulate the customized Blockchain. In our future work, we aim to connect Ethereum Blockchain with the PCA via smart contract.

Chapter 4

The PCA Managed Customized Blockchain Leveraged Decentralized IoT eHealth Framework

The limitation of the work presented in the previous chapter is that the Patient-Centric Agent (PCA) is vulnerable to a single point of failure (SPF) and Denial of Service (DoS) attack due to its centralized architecture. The eHealth architecture includes the PCA which executes on dedicated hardware and participates in administrating a portion of a customized Blockchain. As a result, if the PCA is down, the patient's data will not be recorded in the Blockchain (BC) because the PCA captures sensor's streaming data for transmitting the data to the Blockchain.

The potential safeguard against most cyberattacks is to develop a fault tolerant PCA enabled eHealth system. Faults tolerance[395] refers to a system's ability to continue its service without interruption even if one or more devices or software module fails. A system can be made fault-tolerant by installing multiple hardware or replicating system instances in various devices so that a backup instance of the entire system can be resumed while DoS or SPF attacks disrupt services of the current instance.

Recent technologies such as Fog and Cloud have enhanced the capabilities of the smartphone by accomplishing its heavyweight services. Therefore, instances of eHealth app can be deployed not only on a smartphone but also on Fog and Cloud, which can enhance the reliability and availability of such a system. Instead of saving information to the local drive on a single device, most current IoT applications store information on multiple servers in the Cloud which is a standard virtualized storage and processing system maintained by third party.

However, most current Cloud systems are not designed for meeting all the requirements of IoT data including volume, variety, and velocity of data. Every day, billions of interconnected IoT devices generate more than two exabytes of data[396]. Transferring all IoT data to the Cloud takes huge quantities of bandwidth. Instead of sending vast amounts of IoT data to the Cloud, the most time-sensitive data can be analysed at the network edge, near the user's devices where data is generated[396]. The selected data that requires historical analysis and longer-term storage can be transmitted to the Cloud.

The concept of Fog computing was coined by Cisco in 2014 that describes the decentralisation of the infrastructure of computing or bringing the Cloud resources to user's devices rather than data being transmitted to a Cloud data centre[396]. Fog computing and edge computing, both often termed interchangeably are concerned with exploiting computational resources within a local net-

work to perform computing tasks close to user's devices. The Fog networking comprises a control plane and data plane which facilitates computing resources at the edge of the network. The Fog nodes extend features of Cloud servers and can be mounted anywhere such as on a factory floor, on top of a power pole, next to a railroad track, in a truck, or on an oil rig. The Fog node can be computational devices including industrial controllers, switches, routers, embedded servers, and cameras for video surveillance processing, storage, and network access. Fog computing offers a range of benefits, such as speeding up event response by avoiding an expensive bandwidth to discharge data to the Cloud, and preserving confidential IoT data by analysing it inside the company / user network.

To merge Fog and Cloud technologies with Blockchain technologies, in this chapter, a decentralized Blockchain leveraged eHealth system is proposed where the main innovation involves multiple instances of a Patient-Centric Agent each hosted at multiple layers including smartphone, Fog and the Cloud. The instance of the PCA with all functionalities in a layer can automatically resume its service if a cyberattack stops the main instance of the PCA.

Replicating the same Patient Centric Agent in different layers rather than installing an independent program in each of those layers has several benefits:

1. Secure data flows: Replicated homogeneous PCA instances in three layers can ensure trustworthy, secure, privacy preserving and reliable communication channels between the PCA's at each layer while processing patient data. If one of the PCAs in a layer is under cyberattack, another PCA can take over the role of the compromised instance without sacrificing a patient's privacy. The replicated instances in multiple layers can update their state information at a certain time interval. The PCA in a layer can monitor and analyse requests from other replicated PCAs in other layers. One of the replicated PCAs in each layer acting as master PCA can activate other PCA instances in that layer to step in to perform the compromised PCA role so that if the master PCA experiences flood of responses from malicious nodes or other PCAs of higher or lower layer.

2. Fault tolerance: As mentioned earlier, Fault tolerance refers to a process of leveraging and handling redundancy. The redundancy relates to more resources to accomplish a task than minimal resource to do the task at hand. During failure of a system, redundant resources are exploited to retain normal operations of the system, thereby preserving the desired level of functionality. Four ways of redundancy exist: hardware, software, information, and time[395]. Redundancy of hardware allows additional hardware to be inserted into the design to either detect or circumvent the effects of a failed device. Software redundancy refers to the development of several software instances such that if there are failures in some significant part of the software or down because of cyberattacks, new instances can be mounted. Our concept of replicating PCA in various layers and devices covers redundancy in both hardware and software, which offers fault tolerance capabilities for the system.

3. Blockchain management: Although a PCA architecture contributes to fault tolerance and safeguard cyberattacks, this architecture raises an issue about how multiple PCAs manage the Blockchain to record and process patient data and at which layer the Blockchain can be deployed for maximum efficiency. Fog or Cloud technologies are regarded as an ideal platform to facilitate a distributed peer to peer network for BC technologies. However, both technologies involve several issues while implementing BC with the Internet of Things. The Cloud with virtually unlimited storage and processing capacities experiences an excessive delay in responding to patient's queries.

In contrast, the Fog, which is called an extension of Cloud brings virtual resources close to its affiliated hosts to provide users with the rapid response of their queries. However, unlike Cloud servers, the Fog devices have limited processing and memory capacities which are challenged to run most of the computationally intensive BC consensus protocol and cannot support BC storage. In fact, the BC technologies involve multiple independent operations including structuring transactions, and Block, generating cryptographic keys, executing consensus protocols, and storing confirmed Blocks on a peer to peer network. These BC operations demand a high level of computational, storage and network resources. To address the issues of BC while adopting it in Fog and Cloud, the instances of PCA at different layers are allocated parts of Blockchain operations depending on the layer's processing and storage capacities. For instance, the PCA on the smartphone can define transactions, Blocks, and security keys, the PCA on the Fog can execute the consensus protocol for rapid outcomes related to Blocks and the PCA on the Cloud permanently stores patient data. The BC nodes require the consensus protocol to confirm and insert a Block into a decentralized ledger. The standard consensus mechanism such as Proof of Work (PoW) which is the most decentralized protocol needs excessive computational resources and experiences lower throughput when it is applied to the Fog network.

The Patient-Centric Agent introduced in the previous chapter is shown to minimize energy consumption and increase the throughput of the Blockchain by modifying the Proof of Work consensus mechanism and the structure of Block. However, the Proof of Work consensus protocol cannot execute on Fog level devices due to their high processing power requirements and excessive Block's confirmation latency. Researchers[28] have suggested Practical Byzantine Fault Tolerance (pBFT), PoS, DPoS and other variations of PoS. However, standard PoS and DPoS have limitations, such as nothing at stakes and long-range attacks. Further, these protocols work in a flat network where nodes' variation in terms of processing and storage is reduced, making them unsuitable for IoT deployment.

In contrast, the Fog network comprises of hierarchical heterogeneous low profiling devices including routers, switches, and modems that widely vary in the level of processing and memory capacities. Therefore, the standard PoS protocol does not fit in the Fog network. Considering these facts, we incorporated a lightweight consensus mechanism for Fog devices which is a Fuzzy Inference System (FIS) assisted modified Proof of Stake consensus mechanism in the proposed eHealth architecture.

Since the PCA deployed at multiple layers is hosted by the devices which are equipped with various levels of processing and memory capacities, an instance of the PCA can easily overloaded overwhelmed by the need to process data processing tasks to perform clinical medical tasks. A PCA needs to offload tasks to a host with spare computing resources available to enhance the quality of service in the entire system. However, the preservation of a patient's privacy is particularly challenged while outsourcing tasks to foreign Fog devices in the distributed Fog network because Fog devices are usually manufactured by heterogeneous stakeholders and maintained by third parties. Therefore, an offloading algorithm needs to be designed for preserving privacy of tasks on the Fog network. To address the task offloading issue, a Blockchain leveraged privacy-preserving task migration which assigns the tasks to different Fog and Cloud devices based on the task's sensitivity was designed for a PCA. In this process, Blockchain helps to avoid third parties while processing a task and ensure a tamper-free execution environment.

As mentioned earlier, the Patient-Centric Agent on each layer can outsource some or all of its tasks to other nodes having available resources. The tasks on health data may include a broad range

of services including data compression, filtration, cleaning, billing, key management, designed task migration algorithm accordingly mining, and data analytics related processing etc. The existing state-of-the-art works in task offloading assume tasks are similar. However, tasks on health data vary in the level of sensitivity with respect to privacy, time and energy required.

To bridge this gap, in the proposed task offloading algorithm depicted in Figure 4.1, the tasks are classified as Privacy Sensitive or Normal Task. Each of these types is further categorized as Time Sensitive or Energy Sensitive tasks. Privacy Sensitive tasks are scheduled among the homogeneous Patient-Centric Agent at different layers to ensure patient's privacy and security. Other two kinds of tasks are outsourced to heterogeneous PCA using a Hungarian optimization algorithm.

The tasks that are performed on sensitive medical data such as pregnancy, sexual orientation data are defined as privacy-sensitive tasks. Sensitive data or task is anything that unauthorised users do not have access to. The reason for classifying the tasks before outsourcing to remote/foreign Agents is to preserve a patient's privacy and optimize the migration process with respect to either power consumption or processing time. Exchanging a patient's sensitive medical data with a remote/foreign Agent might risk the patient's privacy. So, the privacy-sensitive tasks of a patient are outsourced to its Agents in the three layers. For other kinds of tasks, two different matrices called time matrix for delay-sensitive tasks and energy matrix for energy sensitive tasks are estimated, and then an optimization algorithm is applied on them separately whereas existing methods formed a single equation or matrix by combining time and energy parameter to compromise both parameters. For instance, real-time surgery, and cardiovascular monitoring patient requires a network with ultra-low latency as a rapid response of these tasks is important in the eHealth system. So, optimizing the scheduling of these tasks with respect to only time is preferable and results in a better outcome. The last issue in designing an efficient task offloading algorithm is to securely collect and maintain execution environment and performance parameters of remote devices. A PCA needs to know the performance parameters of foreign agents including CPU processing rate, memory capacities, location, and network bandwidth to decide which tasks to offload to other agents.

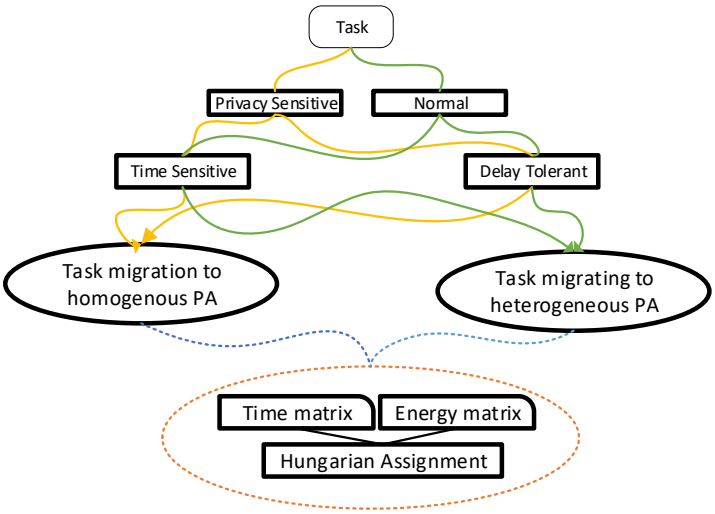


Figure 4.1: The task migration method of the proposed eHealth

Foreign devices may not accurately convey their performance parameters to a Patient Agent intentionally or not. To address this issue, every PCA records their execution and performance

parameters in the BC ledger and updates their execution parameters at a certain interval. The miner nodes verify a remote agent's performance parameters using a special transaction on the Blockchain. Hence, a foreign agent cannot pretend to be a device with high computing resources. Our Blockchain-based decentralized architecture depicted in Figure 4.2 comprises three layers explained below:

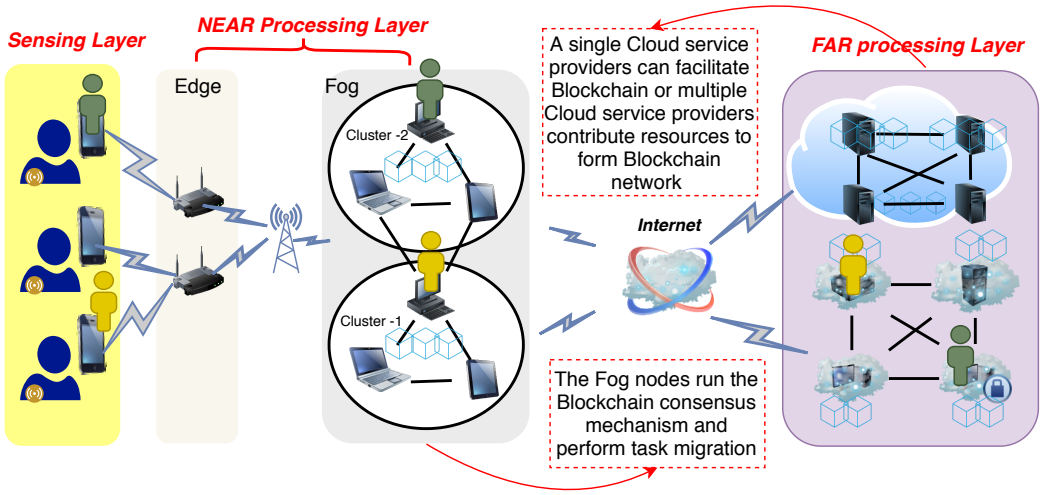


Figure 4.2: The eHealth architecture incorporating the Patient Agent

1. The Sensing layer includes body area sensors and smartphone. The devices in this layer sense data and perform pre-processing on the data before transmitting it to higher layers. The instance of PCA on the smartphone plays the role of coordinating and controlling other instances placed in Fog and Cloud layers. The smartphone Agent acting as a master agent nominates an agent from each layer to monitor the activities of other instances and reports abnormalities of any instances to the master agent.

2. The NEAR processing layer consists of Edge servers, which are at one hop from the sensing devices. The devices in the NEAR processing layer are organized hierarchically. The entire network is divided into several clusters, and one leader or cluster head is elected from each cluster based on several nodes' attributes using a Fuzzy Inference Process (FIS). The FIS makes a set of fuzzy rules using several nodes' features including a node's trust value, performances, and stake in order to nominate a leader from each cluster. The leaders from each cluster participate in collecting and validating transactions produced in its subnetwork. A super leader is randomly chosen from the list of nominated leaders to make a Block, and the node remains active for a certain period or makes a certain number of Blocks. The lead nodes from each cluster verify and validate a Block to be included in the ledger running the upper layer.

   The consensus mechanism is designed to address the shortcomings of the standard Proof of Stake consensus protocol. The proposed consensus protocol applies a cluster-based transactions validation process. As a result, nodes from the entire network are not required to utilize their bandwidth to verify transactions which increases throughput. In standard PoS, miners are selected from the whole network based on the nodes' stake amount. As a result, a particular group of nodes can collude and might repeatedly win mining, whereas our modified PoS allows only one miner from each cluster which prevents BC nodes from colluding and

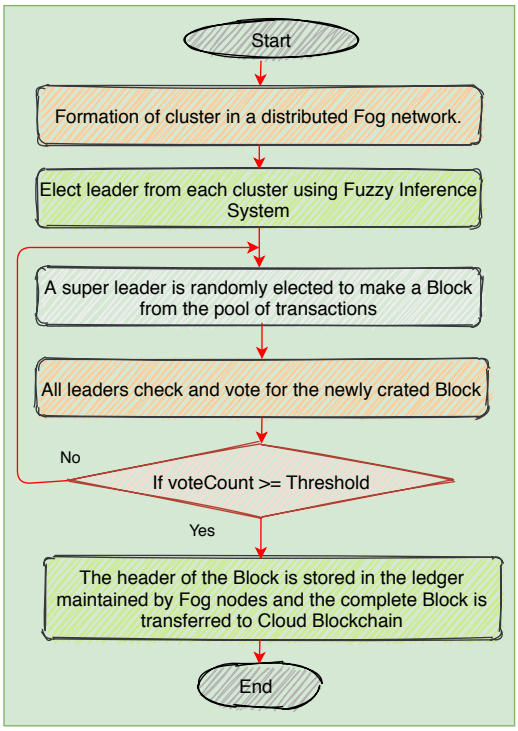


Figure 4.3: The consensus method of the proposed eHealth

makes the protocol more decentralized than the standard PoS. Further, in the proposed PoS, authorities in private or consortium Blockchain can change or set new fuzzy rules or rule's attributes for selecting miner nodes overtime. The modified Proof of Stake methods utilized a Fuzzy Logic System which is flexible and allows the rules to be changed. Such system can accept imprecise, skewed, and mistaken input data. The modified consensus mechanism described above is presented in the flow diagram depicted in Figure 4.3.

3. The FAR processing layer consists of Cloud servers. Multiple Cloud servers from the same providers or different providers form a peer to peer network to maintain the Blockchain ledger. The Cloud servers store health data on the Blockchain ledger and perform heavy-weight tasks.

According to Figure 4.2, the Patient-Centric Agent is replicated at the above three layers with extended features. Each layer can contain more than one replica of the same Patient-Centric Agent to increase the fault tolerance of the system. In addition, we discussed how the Patient Agent could fit on the 5G network to serve various medical services. To secure the framework, each replica of the Patient-Centric Agent shares a data encryption key using Shamir's Secret Sharing (SSS) and utilizes Ring Signature to preserve their privacy.

Shamir's Secret Sharing (SSS)[397] protects a secret key/password in a distributed manner where the secret key is divided into different parts and distributed among multiple entities. A certain number of such shares are required to recreate the original secret key. Similarly, in the proposed eHealth system, a patient's secret key is split into multiple parts, and each part is distributed among the instances of Patient-Centric Agent deployed in each layer. Consequently, no single PCA is in possession of the hidden key. If a replicated Agent is subject to cyberattacks, the compromised agent cannot take control over other instances of PCA because the attacker needs to

compromise other instances too to reconstruct the secret key. Therefore, this approach can preserve the privacy of replicated homogenous Patient-Centric Agents (copy of the same Patient Agent) at each layer.

Most Blockchains like Bitcoin or Ethereum generate a digital signature for endorsing transactions using PKI (public/private key infrastructure) where an owner's transactions are signed by his or her private key. Although PKI can ensure some degree of anonymity for the owner, a malicious attacker can discover the owner's identity because all the owner's transactions are being endorsed using the same set of PKIs. However, a ring signature is considered to be a potential solution to this problem in research.

A ring signature[398] is a form of creating an anonymous digital signature that can be generated by any member of a group. Any individuals of the group can also endorse the message signed with a ring signature. The significant feature of the ring signature is that malicious attackers cannot decide which member's key is used to produce the signature. Hence, the ring signature is suitable for ensuring the anonymity of replicated Patient-Centric Agents. The replicated PCA at different layers form a group for endorsing the ring signature.

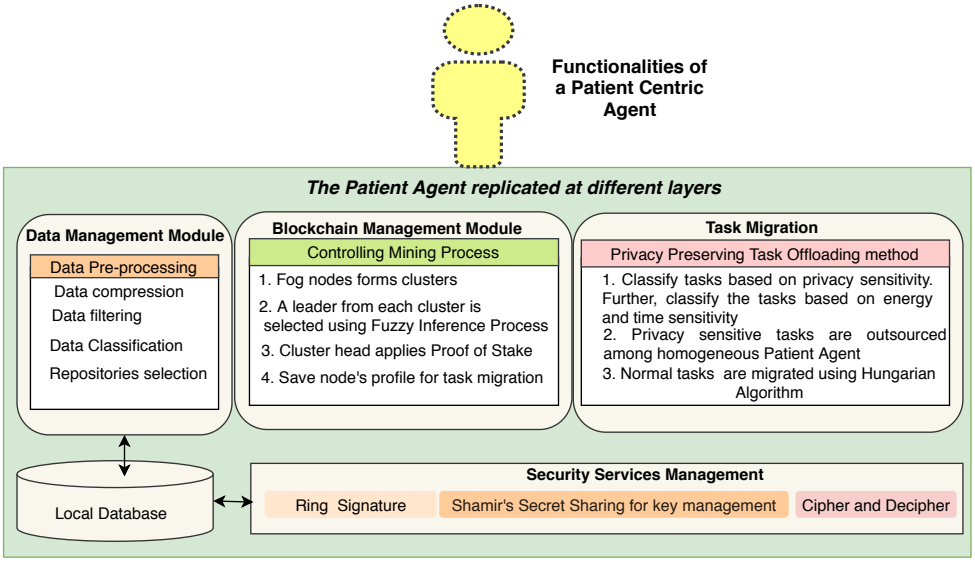


Figure 4.4: The functionalities of the replicated Patient Centric Agent

One of the replicated PCAs builds ring signatures on the patient's transactions. The digital signature also contains other agents' public key. Therefore, other PCAs outside this group cannot discover who belongs to the transaction and the digital signature (A foreign agent cannot parse the digital signature to find which member's public key is in the signature to discover the owner agent's private key). Figure 4.4 presented the functionalities of the Patient-Centric Agent replicated at three layers. The PCA has four main modules: Data Management Module (DMM), Blockchain Management Module (BMM), Task Migration Module (TMM) and Security Service Module (SSM). The DMM handles pre-processing of sensors' data and determines the health data storage repositories. The BMM performs Blockchain-related operations such as mining, building, and endorsing transactions. The TMM decides to migrate patient's tasks and choose remote/-foreign agents for outsourcing tasks if necessary. The SSM encrypts or decrypts patient's data, produces digital signature and update security key. Finally, the performance of the framework is analyzed by simulating the key algorithms using Java Programming. The security protocols are simulated on Scyther tools[129] to measure their strengths against cyberattacks.

The contents below of this chapter were published in Internet of Things Journal, Elsevier in January 2020. The article has already been cited by 18 times (according to Scholar Google).
M. A. Uddin, A. Stranieri, I. Gondal and V. Balasubramanian, " Blockchain Leveraged Decentralized IoT eHealth Framework," Internet of Things, ELSEVIER, vol. 9, pp. 100159, 2020. https://doi.org/10.1016/j.iot.2020.100159.

Abstract

Blockchain technologies recently emerging for eHealth, can facilitate a secure, decentralized and patient-driven, record management system. However, Blockchain technologies cannot accommodate the storage of data generated from IoT devices in remote patient management (RPM) settings as this application requires a fast consensus mechanism, careful management of keys and enhanced protocols for privacy. In this paper, we propose a Blockchain leveraged decentralized eHealth architecture which comprises three layers: 1) The Sensing layer- Body Area Sensor Networks include medical sensors typically on or in a patient body transmitting data to a smartphone. 2) The NEAR processing layer- Edge Networks consist of devices at one hop from data sensing IoT devices. 3) The FAR processing layer-Core Networks comprise Cloud or other high computing servers). A Patient Agent (PA) software replicated on the three layers processes medical data to ensure reliable, secure and private communication. The PA executes a lightweight Blockchain consensus mechanism and utilizes a Blockchain leveraged task-offloading algorithm to ensure patient's privacy while outsourcing tasks. Performance analysis of the decentralized eHealth architecture has been conducted to demonstrate the feasibility of the system in the processing and storage of RPM data.

4.1 Introduction

Recently a wide range of IoT health applications has been developed to automate and make health services accessible to individuals. For instance, remote patient management (RPM) covers a variety of health services, including continuous vital signs monitoring with wearable or implantable sensors, arrhythmia detection, fall detection, oxygen therapy regulation, monitoring of pregnant women, chemotherapy reactions, and glucose monitoring[91].

However, IoT services have not flourished to the extent expected due in part to challenges associated with reliability, fault tolerance, and privacy challenges. In eHealth, patient's physiological data, captured by medical IoT(Internet of Things) devices is transmitted to Edge or Cloud entities managed by third parties, which makes data security and the preservation of a patient's privacy challenging.

Most IoT systems are centralized in that data flows to a single server for processing and storage. This makes these systems vulnerable to a single point of failure, particularly while handling a large number of end-to-end communications [82]. Cyberattacks such as Ransomware, and Denial of Service(DoS) attacks can paralyze conventional eHealth systems and dangerously disrupt healthcare services [399]. Recently, health data has become more attractive to attackers; the US Department of Health and Human Services(HHS) reported over 2250 data breaches between 2009 and 2018[400]. Further, insiders such as healthcare professionals, support staff, and service providers are associated with over half of recent health data breaches[400].

A typical IoT architecture involves a single instance of a health app maintained on a smartphone transmitting data to Edge devices then Cloud servers which are managed by third parties. This

architecture is vulnerable to eavesdropping attacks over Bluetooth, Zigbee, or WiFi links, a man in the middle attack, DoS, and insider attacks. Further, as outlined, conventional Edge[92] or Cloud process[84] cannot guarantee accountability and tractability of patient's data due to third party involvement.

The processing of data in the Cloud in the typical IoT architecture requires a high level of accountability[401] and transparency over how data in the Cloud servers are used. Statutes and regulations for the regulation of health data now exist in many jurisdictions; however, Cloud service providers compliance is difficult to ascertain. This undermines the trustworthiness of Cloud-based processing of data.

The mobile to Cloud component of the IoT architecture called Mobile Cloud Computing(MCC) [85][86] has extended the capabilities of a smartphone by enabling uploading health data to Cloud server for processing and storage. However, the efficacy of MCC depends on the extent to which connectivity loss and latency can be minimized. Without this, the large storage and processing capacity of Cloud environments, cannot be utilized because of excessive transmission delays and unstable connections.

Multi-access Edge Computing (MEC)[402], also known as Mobile Edge Computing, provides an IT service environment at the edge of the cellular network and closer to the customer to access Cloud computing capabilities. MEC supports distributed edge computing by processing content on Edge devices such as base stations, radio network controllers, hot spots, local data centres, routers, switches, and WiFi access points. Collecting and processing data closer to the customer reduces latency and brings real-time performance to applications requiring high-bandwidth. Most Edge computing initiatives are being developed using open-source hardware and software that leverage Cloud and virtualization paradigms, including SDN(Software Defined Network) and NFV(Network Function Virtualization). The MEC platform supports multi-tenancy for cellular operators to rent their radio access network to authorized third parties such as application developers and contents providers. In distributed Edge network component of MEC, an Edge server with lower processing capabilities can outsource its tasks to remote Edge servers with higher computing capabilities. But, migrating tasks to remote Edge servers introduce security threats that can result in data theft, and privacy breaches because diverse stakeholders manage Edge computing devices.

In this article, a Blockchain is deployed to perform task migration on the Edge network. A Blockchain consensus mechanism is executed on the MEC to provide its tenants and customers with faster processing and higher security. This adaptation to IoT architectures and MEC is very important for dealing with health data where privacy concerns, the need to trust providers and quality of service demands are very high. This is particularly true for patient-generated health data.

Patient-Generated Health Data(PGHD)[399] such as biometric data, symptoms, lifestyle choice, and treatment history collected through sensors, mobile apps, web protocols, and home monitoring devices are distinct from healthcare data within a clinical setting. Ideally, a patient has control over how and where their patient-generated data is stored and shared. Patients may need to share their generated data with diverse health care providers who each, typically maintains their own electronic medical record[403]. Patients may also choose to include their generated data to electronic health record(EHR)[404], an inclusive record of patient medical history, more extensive than medical records, to be shared and accessed by authorized users from across different healthcare providers.

An eHealth system is required to process, cleanse, analyze, and manage the patient-generated data to ensure it is accurate, complete, accessible to authorized users, and understandable to healthcare providers and meets patient's Quality of Services(QoS) expectations. These tasks require con-

siderable computational resources, and cannot be achieved without compromising performance or security aspects of QoS expectations. Recently researchers have applied Blockchain technology to address privacy, security and third parties trust concerns regarding the management of medical data and healthcare services. Blockchain[73],[405] can support access control, secure storage and sharing of medical data without the need to trust third parties or intermediaries while maintaining user's privacy. Blockchain-based ehealth architectures [54, 89, 91] have been designed to manage health data autonomously. Further, distributed ledger technology, popularly known as Blockchain, has also been investigated to manage user's identity[406], metadata [**?**], share, and maintain logs of medical records[407] and patient key management[408][409]. For instance, MedRec[407], a prototype of a decentralized Blockchain architecture, was first implemented to contribute to an interoperable EHR system. In this eHealth system, Ethereum smart contracts orchestrate contents across different storage and provider sites. The system facilitates a comprehensive medical record view, care auditability, and data sharing through governing authentication logs. Linn et al.[**?**] suggested an off-chain to manage raw health data and Blockchain to store metadata of medical records to tackle the challenges of accommodating decentralized ledger technology in healthcare.

However, the adoption of Blockchain in healthcare has mixed reviews. Inclusion of Blockchain technology in healthcare is a trade-off between security and computational cost and storage resources.

Execution of Blockchain's algorithms, including mining, authorizing creates a burden on resource-constrained devices such as medical sensors and other IoT devices. The underlying core component of a Blockchain, the consensus mechanism refers to a common agreement amongst Blockchain Miners about the Block's state. Consensus mechanisms applied in various Blockchain-based eHealth research require very high computational resources. For instance, the Proof of Work consensus mechanism applied in [54, 89, 90] needs high computational overhead, long delays, and a great deal of power. Dwivedi [91] attempted to reduce the computational overhead on IoT devices by deploying a Gateway node to gather data Blocks from a group of IoT devices. The Gateway verifies the Blocks as a Miner before adding the Blocks to the Blockchain overlay network. Tuli et al. [90] presented a generic Broker in the Fog to adopt Blockchain into the Internet of Things data streams. The broker assigns Blockchain tasks to various Fog devices so that the computational challenges can be met. However, these approaches are still vulnerable to DoS attacks and tampering because they still rely on a centralized Blockchain controller. Fault tolerance capabilities of eHealth architecture have not been substantially addressed. Fault tolerance[410] refers to the capability of a system to continue its operation or keep delivering uninterrupted services even if its significant components or partial components stop working. A system can be made fault-tolerant by adding a software module that continuously monitors the activities of functional units or replicating the system among multiple hardware.

The approach advanced here involves the creation of multiple instances of a Patient Agent which is a software module dedicated for a patient. The multiple instances of the Patient Agent reside in three levels; patient's smartphone, Fog devices and Cloud servers.

However, the advancement of digital health increases the demands of various medical data services such as advanced user-centric applications at an affordable price, context-aware and proximity services, service delivery crowded areas, and advanced multimedia centric services. The vision of future 5G system can provide the customer with the above outlined customized and advanced healthcare services. 5G network has promised to provide faster speeds and more reliable communication connections on smartphones and other devices. 5G network is defined more beyond offering low latency and ultra-high bandwidth to services. 5G has brought the concept of network slicing into reality with the help of Software Defined Network(SDN) and Network Function Virtu-

alization(NFV) technologies. Network slicing creates multiple isolated end-to-end virtual/logical networks on top of a shared physical network[411] to meet the resource requirements for diverse kinds of applications. Each logical network possesses dedicated and shared resources in terms of processing power, storage, traffic capacity, connectivity, and coverage latency, and adequate bandwidth required for an application[412]. 5G technology is envisioned to support a wide range of applications, including IoT, vehicular network, and healthcare[413]. Healthcare has a diverse range of applications such as mobile health, assisted driving, aged care. These applications require various levels of resource requirements(processing power, storage, and bandwidth). For instance, an emotion-aware application supporting real-time emotion detection should operate on a network slice, offering low latency to ensure the patient's QoE(quality of experience). In remote surgery applications, a network slice with ultra-low latency and highly reliable connections are required to ensure the safety and security of the patient. In this article, we describe how the Patient Agent and the Blockchain can be envisaged on the 5G network to manage and allocate logical resources to diverse health applications to improve Quality of Experience(QoE).

In this paper, we focus on developing a Blockchain leveraged eHealth framework. The main contributions of this paper are summarized as follows:

1. An eHealth system leverages Blockchain technology with an architecture consisting of three layers; the sensing layer(Body Area Sensor Network), NEAR processing layer(the Fog) and FAR processing layer(the Cloud). Multiple instances of the Patient Agent named smartphone Agent, Fog Agent and Cloud Agent are deployed at three layers to process patient's data.

2. A customized Blockchain with a lightweight modified Proof of Stake consensus mechanism implemented at the NEAR processing layer to process data streamed from medical IoT sensors. The consensus mechanism for the healthcare Blockchain is executed on the Edge devices to best use those device's faster communication capacity leaving the permanent storage for the Blockchain to be managed in the Cloud.

3. Proposal of a Blockchain leveraged Task Migration Algorithm. The method migrates tasks to neighbouring or remote Fog Agents based on data sensitivity where remote Agent's profile is managed on the Blockchain.

The rest of the article is structured as follows. We review related research in Section 7.2 and describe our proposed eHealth system in Section 7.3. The performance of the proposed approach is presented in Section 7.4. The security analysis of the architecture has been performed in Section 4.5 before concluding the article in Section 7.5.

4.2 Literature Review

In this section, we review literature in four categories: conventional healthcare architecture, Blockchain-based healthcare architecture, 5G enabled eHealth and Blockchain consensus mechanism.

4.2.1 Conventional Fog/Cloud Healthcare Architecture

Recent eHealth architectures [87], [88],[85] incorporated Fog computing with smartphone and Cloud to make the processing of health data faster. Mahmud et al. [87] presented an IoT eHealth

system where pre-processing including data compression, filtration and analytics are performed on the Fog devices before transmitting data to Cloud servers for advanced processing. Rahman et al. [88] advanced an eHealth prototype by exploiting a smart gateway at the Edge network. The smart gateway implemented using UT-Gate supports a good range of high-level services, including local storage, real-time local data processing, early warning, and embedded data mining. The health data processed by the gateway is transmitted to the Cloud servers for future access.

Verma and Sandeep[414] suggested an eHealth architecture with a feature of diagnosis. A gateway or local processing unit residing at the Fog layer receives health data from medical IoT devices to perform pre-processing. They advanced a disease diagnosis module in the Cloud subsystem. The module generates alerts and warnings for caregiver subsystems. Fernandes et al. [415] designed a multi-agent-based remote patient monitoring system where the roles of the agents cover a wide range of activities including identification, collection, storage, recovery, visualization, monitoring anomalies, resource notification and dynamic reconfiguration. The authors also constructed IoT4Health as an eHealth solution. Jin et al. [416]'s eHealth system integrated IoT devices and Cloud services. The IoT devices transmit health data to a mobile access point(MCP) to upload the data in the Cloud. The MCP chooses a Cloud service provider while optimizing bandwidth and maintaining the service deadline. Aazam et al. [85]'s eHealth system included a resource provisioning technique in the Fog. The technique used the relinquish probability model to reduce the wastage of resources of edge layer devices.

4.2.2 Blockchain Based Healthcare Architecture

Although the aforementioned eHealth frameworks have explored Fog/Edge computing for rapid access and processing of medical data, patient's security and privacy are not addressed in those proposals. Both Fog and Cloud involve their administrators to process medical data which threatens patient's privacy. Blockchain implemented over Fog and Cloud might enable the processing and storage of patient data while avoiding the reliance on the centralized Fog/Cloud administration. Blockchain structure has motivated researchers to devise privacy-preserving eHealth systems.

Uddin et al.[54] proposed a Patient-Centric Agent residing in the Smart Gateway to determine storage, access control and privacy level during the insertion of patient abnormal medical data into a customized Blockchain. The Patient-Centric Agent also selects Blockchain providers to schedule medical data for processing and storage. In [89], the role of the Patient Agent is extended to manage multiple Blockchains and multiple storage mediums, including Local Computer, Cloud to preserve patient's privacy. Tuli et al.[90] presented FogBus that is a lightweight Blockchain-based Fog computing framework. They introduced a universal broker software executing on the Fog device to merge Blockchain with Edge devices such as medical sensors. The broker schedules jobs among other devices in the Fog. However, a universal broker system in eHealth causes security and privacy threat for the patients. Rahman[417] proposed a secure therapy framework, including Blockchain at MEC(Mobile Edge Computing) and the Cloud. The therapy data from physician and patients are processed by Cloud and MEC Blockchain nodes to ensure immutable, anonymous, secure and transparent sharing. The Blockchain stores only hashes of the therapy multimedia and the actual multimedia data containing images, audios, videos are stored off-chain in a separate database. Although the framework includes MEC Blockchain to avoid shortcomings of high bandwidth and analytical processing required by the Cloud, the Ethereum consensus consumes high power at MEC.

Griggs[35] presented an architecture for automated remote patient monitoring using smart contract of the Ethereum. A smart device such as mobile or laptop collects and aggregates data trans-

ferred by body area sensors. The smart device sends the aggregated data to pre-specified smart contracts stored on Ethereum. The smart contract processes the data and sends the result and notification to smart devices and healthcare providers. The Blockchain only keeps a record of the event's occurrence, and data is stored on the Electronic Health Record(EHR). However, the smart devices can cause a single point of failure and be vulnerable to Denial of Service attack. The architecture only ensures the secure processing of medical data. Chen et al.[171] discussed a Blockchain-based medical data access framework, including the Cloud to store medical data. All kinds of communications between a patient and the third party insurance or healthcare professionals are logged in the Blockchain. Liang[418] developed a Blockchain integrated personal health data sharing framework where a mobile app gathers data from wearable sensors and inserts data into the Cloud and Blockchain hyperledger to verify the integrity of the data. Zhang[84] applied Blockchain-based architecture called FHIRChain to securely and scalable share clinical data. The requirements defined by "Shared Nationwide Interoperability Roadmap" were focused in the architecture. Brogan[419] described the role of distributed ledger technologies to advance the electronic health record, ensuring the authenticity and integrity of health data. The authors also demonstrated the application of IoTA protocol masked authentication messaging extension module for securely sharing, storing and retrieving encrypted data using a tamper-proof distributed ledger. Gordon[420] discussed the facilitation of Blockchain in patient-driven or patient mediating data interoperability with respect to health data accessibilities, aggregation, liquidity, identity and immutability. Rupasinghe[421] identified two categories of risk factors of fall, namely medical factors and environmental factors. All identified risk factors are labelled as weak, moderate and strong based on evidence and expert opinions. Four types of users put fall-related data to a consortium Blockchain to ensure the interoperability, accessibility, and availability of the data to predict the likelihood of fall of the aged people. The smart contract is proposed to perform user's registration, insertion of data into Blockchain and fall prediction.

Dwivedi et al.[91] presented a Blockchain oriented eHealth frameworks inspired by Ali et al.'s proposal of a lightweight Blockchain for IoT [287]. They considered an overlay network which is a peer-to-peer computer network built on top of another network where nodes are logically or virtually connected. In the proposal, the IoT medical devices generate Block to be verified by the cluster head of the overlay network before sending them to the Cloud servers. The authors utilized several lightweight standard security protocols to enforce security and preserve patient's privacy in the eHealth system. A static cluster head always verifies the integrity of the Blockchain. However, avoidance of a global consensus mechanism can weaken the sustainability of Blockchain-based eHealth system. We advanced our eHealth by running a lightweight consensus mechanism on a peer-to-peer Fog network. The cluster head nominated for a certain period based on nodes' properties execute the consensus mechanism. Further, the functional blocks required for monitoring a patient is bundled in the form of a Patient Agent (PA), and the PA is replicated in devices at three levels: smartphone, Fog and Cloud levels.

Gaetani[422] proposed a Blockchain comprised of two layers in the Cloud computing environment. The Blockchain in the first layer keeps a record of operations issued on the distributed database while avoiding the computationally expensive Proof of Work. The Blockchain in the second layer records the logged operations generated from the first layer database using Proof of Work. Novo[423] designed a Blockchain-based decentralized architecture to manage access control for memory and power-constrained IoT devices. The key feature of the architecture includes a manager hub between IoT devices and Blockchain. Manager hub requests Blockchain node for access policies stored in the Blockchain for a particular wireless sensor network. The smart contract is executed to insert access policy into the Blockchain.

Existing Blockchain eHealth architectures reviewed above included Fog and Cloud for the storage and processing of patient's data. However, those studies related to healthcare did not advanced the notion of executing Blockchain's controller in a distributed fashion at multiple levels like sensing, near and far processing levels. Decentralized Blockchain controller makes an eHealth system fault-tolerant, reliable and protects it from the DoS. Further, Blockchain based existing eHealth architectures did not advance consensus mechanism and privacy aware task offloading method. To bridge this research gap, we proposed a decentralized Patient Agent-based eHealth architecture with Blockchain implemented at Fog and Cloud level described in the next section. Here, a comparative analysis of Blockchain-based health records and conventional health records systems is presented in Table 4.1 and 4.2.

4.2.3 State-of-the-art 5G enabled eHealth systems

Some 5G enabled eHealth systems have recently been developed to support various personalized human-centric interactive applications. Chen[428] proposed a MEC (Mobile Edge Cloud) based emotion-aware architecture using the features of 5G networks such as high data rate, low latency and high computing capacity. In this proposal, mobile devices collect emotion-related information and send information to Cloudlet and remote Cloud for processing. LIN et al. [429] designed a system to handle big data in emotion-aware application using SDN and 5G technology. They described the functionalities of data collection, transmission and storage for the emotion-aware application. Data is transmitted to the control layer of SDN(Software Defined Network) with high throughput capacities and then forwarded to the data centre via SDN application. Finally, data is also uploaded in the Cloud for analysis and storage. Hossain et al.[430] also presented a 5G enabled framework to recognize and monitor emotion using speech and image. Authors added a component to recognize emotion in a 5G based cognitive healthcare framework. They extracted features from the captured image using local binary pattern (LBP) and interlaced derivative pattern (IDP). The structure incorporated Bluetooth technology so that caregiver can estimate the precise location of the client.

Chen[413] presented a healthcare architecture focusing on data-driven computing and caching in 5G networks. Chen included an SDN based resource cognitive engine that optimally allocates resources analyzing patient's need in diverse situations. Further, they introduced caching called small cell Cloud existing in the mobile and Fog devices and macrocell residing in the Cloud to provide the user with better QoE(Quality of Experience) in a 5G network. Chen[431] also proposed a 5G enabled smart diabetes system with analysis of diabetes patient's suffering using a machine learning method. A social networking based data sharing model is also presented for 5G smart diabetes.

Sharma[224] presented a Blockchain oriented distributed SDN controller architecture for IoT networks. Traditional SDN controller is centralized. Sharma proposed to use a distributed SDN controller that would be connected using Blockchain technology. This architecture can withstand major cyber-attacks, whereas centralized SDN controller is vulnerable to a single point of failure. However, the incorporation of Blockchain at SDN controller level might have delay and additional computational cost in processing of the user's data using API(Application Programming Interface).

State-of-the-art research focused on the design of 5G enabled distinct health applications. However, a patient might need services from more than one health applications at a time. For instance, a patient might require both 5G enabled emotion detection and diabetic monitoring applications at the same time. Therefore, there needs a dedicated module to manage and control health applications personalized for a particular patient and determine resource requirements for those patient's

Table 4.1: The comparative analysis of conventional healthcare system and Blockchain based healthcare system

Parameter	Conventional Healthcare	Blockchain Healthcare
PIA(Privacy, Integrity, Availability)	Created and managed by healthcare professionals or national government or patients themselves where record managers can access a patient's record without the patient's knowledge. Health data is stored in off-premise third party providing servers such as Cloud server. Consequently, data integrity cannot be guaranteed[83] in these systems. Operational failures can make health services unavailable.	Patient-driven health record management system. Privacy is at risk if an attacker can correlate record transactions to the patient by analyzing the transaction's contents[367]. Blockchain Consensus mechanism ensures data integrity. Replication of complete health record amongst multiple entities can guarantee uninterrupted health services.
Freshness	Attackers might manipulate local timestamp of a centralized server.	Timestamped Block is added to the Blockchain after Miner's verification and the attacker cannot alter a Block's timestamp
Cyber Attack	Dissemination of patient records has real-world consequences, including diverse cyber-attacks: DoS, ransomware. Various restrictions are imposed on sharing massive health data in conventional EHR due to the risks of leakage of personal information[424]	The Blockchain healthcare has been proven to safeguard the system from DoS, ransomware but also suffers from some other attacks depending on the Blockchain algorithm such as long-range attack, mining and storage attack and known 51% attack.
CAP(Cost, Access, Performance)	Involvement of high deployment, maintenance and administrative costs[425]. Access is delayed due to fragmented health record among diverse health record systems	Government or stakeholder does not require a massive amount of deployment cost since individual entity contributes resources. Instead, cost-saving and profitable because a user is rewarded for taking part in mining. This can alleviate employee wages, legal cost, and data centre rentals [426] but outputs low throughput and high power consumption depending on the consensus mechanism.

Table 4.2: The comparative analysis of conventional healthcare system and Blockchain based healthcare system

Parameter	Conventional Healthcare	Blockchain Healthcare
SI(Standardization, Interoperability)	Maintenance of standardization across the various agencies[427]. Diverse health institutions or providers may have various legal requirements which add an extra barrier to the cross-border sharing health data. Different security and privacy methods used in such systems raise the interoperability issue among them.	These systems are lacking high-level standardization which obstructs the fast development of decentralized ledger technology in health sector[83]. But, universal set of rules and regulations of such technology can offer a high level of interoperability to share records across diverse institutional healthcare professionals.
AT(Auditability, Trust)	Logging information for every access, but auditing is not transparent due to the involvement of a single controller or institution. Resilience and sustainability of such a system rely mostly on single third party	Data is recorded in an immutable distributed ledger with auditing traces which guarantees transparency in the procession of data exchange [424]. Every entity maintains logging. Transparent sharing of health data by running a consensus mechanism, which substitutes the third party trust
Tamper-proof storage	Public or private Cloud generally handles the storage and processing of Big health data [161]. These repositories are unable to guarantee immutability of health record	Digital signature, the cryptographic linkage between Blocks, and a consensus mechanism make reliable and fault-tolerant storage. The Blockchain ledger replicated among multiple servers prevent a record from being tampered.[161].

health applications.

4.2.4 Consensus Mechanism in Blockchain

Gai[432] presented a Proof of Reputation based consensus mechanism for a peer-to-peer Blockchain network where a service provider receives some reputation from service requestors as incentives. A group of Miner is selected based on their earned reputation. The publications of their reputation transactions are blocked by the system instead of digital coin if a Miner's malicious activities are detected by the system. However, only reputation revoking cannot prevent Miner from performing malicious activities on the Blockchain. Further, some nodes can collude to work for the requestors and obtain high reputation.

Li[433] proposed a Proof of Vote(PoV) consensus mechanism where a group of commissioners vote for the selection of butler who are responsible for mining on the Blockchain. PoV outputs higher performance in terms of power and delay. But PoV is not fully decentralized like Proof of Work rather it provides controllable security, convergence reliability, and only one Block is confirmed within a time frame.

Bitcoin-NG proposed by Eyal et al. [434] selects a Miner through broadcasting macro Block on the Blockchain. The node that comes up with target hash(Proof of Work) for the micro Block mines the data Block. The Bitcoin-NG can reduce the latency and power consumption required for broadcasting data transaction throughout the network. Peterson et al.[435] followed Multi-Chain[436]'s PoW to randomly select Miner to perform validation on the Blockchain. Random Miner selection improves the throughput and reduce computational overhead. However, malicious and inefficient Miner might be nominated as Miner in random selection process. A hybrid consensus mechanism including reputation, voting, performance, randomness and stake of coin can meet challenges related to consensus protocol.

4.3 Decentralized Patient Agent based eHealth Framework

In this section, our eHealth framework is described in detail. Figure 7.2 depicts the high-level view of the eHealth architecture that comprises devices at three levels: Sensing, NEAR processing and FAR processing level. Devices at three levels contain multiple instances of a Patient Agent. The Patient Agents at the upper two levels: NEAR and FAR processing layers collaboratively take part in realising Blockchain technologies to process patient's data securely. An Agent dedicated to providing a patient with health services has several functionalities, including handling task migration, managing Blockchain, and network slices. The Agent's **Migration Handler(MH)** decides to migrate a task or locally executes the task using **Profile Monitoring(PM)** that collects profile information of remote Agents from the Blockchain. The **Execution Unit(EU)** locally processes a patient's health data. The **Blockchain Manager(BM)** of the PA assists EU, MH and Storage Management(SM) to accomplish their activities through a consortium Blockchain. Consortium Blockchain(CB) is not granted to a single entity like private Blockchain. Instead, CB, a semi-decentralized system, is managed and controlled by a group of approved entities. The details about PA's functionalities depicted in Figure 7.4 are discussed in the later section. The three layers of the eHealth system are discussed below.

- **The Sensing Layer:** Wearable sensors, implantable sensors, smartwatch, smartphone and other IoT devices sense patient's vital signs and passive data, including room temperature,

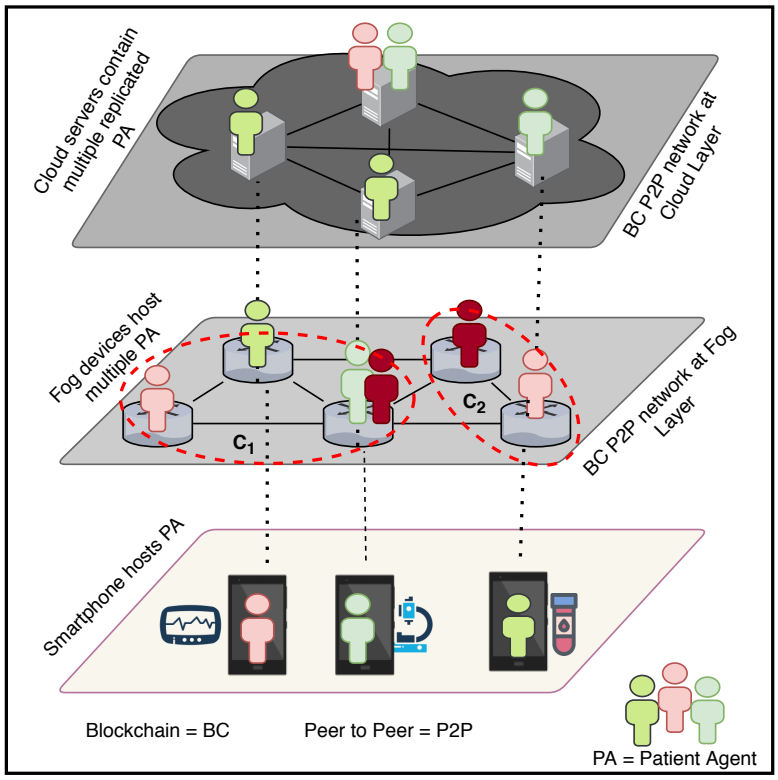


Figure 4.5: The eHealth architecture incorporating the Patient Agent

humidity. The IoT devices at the sensing layer are wirelessly connected to a smartphone via star topology. These devices transmit a patient's physiological sign, including ECG, EEG, BSC to the smartphone using Bluetooth or ZigBee protocol[364]. The smartphone instance of a PA performs pre-processing such as filtering noise, classification on physiological data to send them to the next level for further processing.

- **NEAR Processing Layer:** The devices of NEAR processing level are typically located at one hop distance from the data sensing devices. This NEAR processing layer also called Edge computing network, comprises traditional switches, router, and low profile devices [88]. The devices in the NEAR processing level form a peer-to-peer network. A modified Proof of Stake(PoS) consensus mechanism executes on this peer-to-peer network. Each Fog device runs the similar suits of Blockchain protocols. As a result, they can perform communications between them without the need to trust third parties[82].

- **FAR Processing Layer:** The FAR processing level includes servers with high computing and storage capabilities. The location of these devices can be far away from the data sensing devices. Cloud servers managed by various proprietary organizations like Amazon, Microsoft, IBM and other stakeholders can provide servers with a large volume of storage and high computing capabilities. Blockchain maintains a distributed tamper-proof ledger replicated amongst multiple nodes. Managing health data characterized as Big data are challenged with the storage in decentralized distributed ledger.

  The Cloud servers [437] might support massive storage required for handling decentralized distributed ledger for health data. Multiple instances of a Patient Agent are also deployed in

the Cloud to process delay-tolerant and high computing tasks with greater availability and flexibility.

The NEAR and FAR processing layers, each contains n ≥ 2 number of instances of a PA and the sensing level has at least n ≥ 1 number of PA instances. The Patient Agent hosted on the sensing layer can randomly choose a PA from the NEAR or FAR processing layer every time it has health data to be processed in the upper layers. The smartphone Agent on the sensing level plays the role of master Agent and instructs, one of the NEAR Agents to monitor other NEAR and FAR Agents. The monitoring Agent reports the master Agent if malicious attacks occur on a NEAR or FAR Agent. If an Agent is shut down due to cyber-attacks or suffers from network overload, the master Agent activates a new Agent to take over the role of the infected Agent.

4.3.1 5G Network Architecture

Logical network slice in 5G technology instantiates on a standard network that comprises terminal access network, core network, access network and transport network. Network resources in 5G are typically defined using two terminologies: infrastructure provider(InP) and tenant. Each InP rents its virtual network resources to the tenant following the business service rules and agreements. The InP can usually host multiple tenants. The tenants can be an infrastructure provider for other tenants. Tenants manage network slices in accordance with the resource demand from different services or applications. Figure 4.6 depicts the concept of InP and tenant. In Figure 4.6, three physical networks have been provided by three infrastructure providers(InP). Virtual resources on top of each physical network are allocated to a variable number of tenants. The tenants facilitate network slices to run a wide variety of applications.

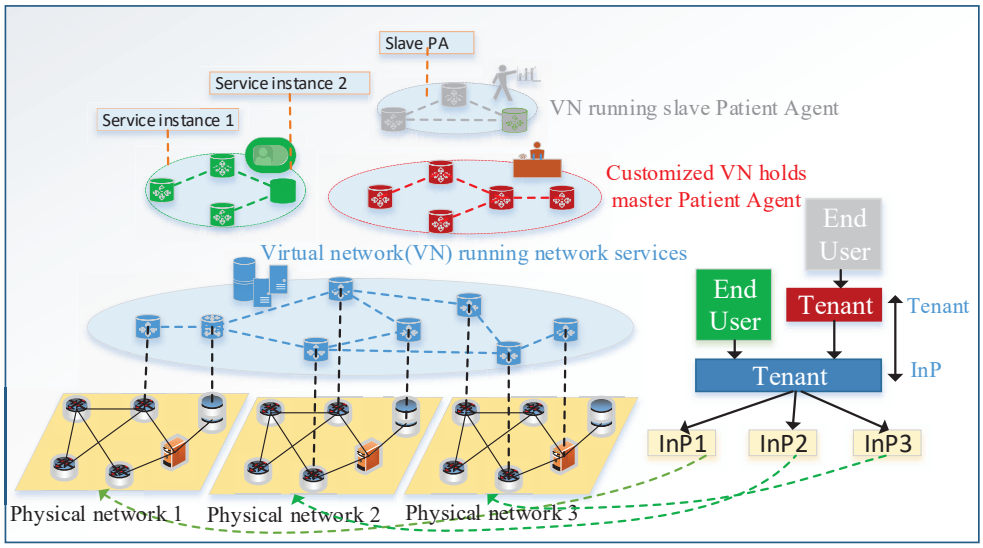


Figure 4.6: The 5G network supporting multi-tenancy

Here, we first described a 5G slice architecture designed by ETSI[412]. Next, we explain how the Patient Agent might operate on this 5G network architecture. Figure 4.7(a) suggests that the resource management in 5G is accomplished in two levels- NFVI level and tenant level. NFVI (Network Function Virtualization Infrastructure) refers to virtualization of hardware resources including computing, storage and networking. Virtual network functions (VNF) that entail distinct

network functions such as routing, intrusion detection, domain name service (DNS), caching, and network address translation (NAT) run on these virtual resources.

In NFVI level, VIM directly manages virtualized resources instantiated on the underlying hardware in the form of VMs. VIMs delegate managerial tasks related to network resources to their underlying ICs. The ICs followed by the VIM programmatically manage the network resources and support VM connectivity at the virtualization layer. Like VIM, WIM(Wireless Interface Manager) manages virtual resources related to forwarding instructions or transport layer via WAN IC, which is the SDN controller at NFVI of the transport layer.

SDN (Software Defined Network) controlled Tenant is located on the top of NFVI that provides a tenant with virtual resources. The tenants on top of NFVI manage a set of network slices. Each slice consists of an OSS, a TC(software-defined network controller) and an NSO. The OSS is an SDN application from TC's perspective. The OSS commands TC to create a slice's constitutes(VNFs) and logically compose them to realize the network services. The NSO regulates the life cycle of network services and interacts with the TC via OSS. The TC organized as a VNFs depends on the capabilities of the virtual routers, and switches. The TC has two interfaces: northbound and southbound, to interact with end-user and forwarding instructions, respectively. The TC uses southbound interfaces to send composition and forwarding pertinent instructions to virtual routers and switches. The northbound interface of the TC enables users to solicit the required resource capabilities for the network services that they select. Further, the northbound interface enables end users to manage, and operate network slice within limits set by the tenant and retrieve context information such as real-time performance, fault information, and user policies regarding network slices.

The RO, a functional block of a tenant, orchestrates its assigned resources from multiple NFVIs to dynamically satisfying the diverging requirements of network slices. The RO provides each slice with required resources via interfaces of each slice's NSO. The RO(Resource Orchestration) at the tenant level is connected to VIMs of different NFVIs to deliver its assigned resources to the corresponding slices. The tenant can access, reserve, and request virtual resources through RO. VIM and WIM from different NFVIs can interact between them via RO.

4.3.2 The Role of the Patient Agent on 5G network

A patient might suffer from multiple complexities and diseases during his or her life span. Handling different medical cases require network slices with various levels of resource capacities to meet patient's QoE and QoS. For instance, resource requirements to serve a patient with diabetes differ from the resource requirements to monitor a patient who suffers from arrhythmia. Therefore, a personalized software Patient Agent is required to continuously record, and analyze health data to assess resource requirements for offering appropriate health services to a patient.

The Patient Agent can fit on the 5G architecture described in section 4.3.1. We propose to incorporate a master Patient Agent with OSS(Operation Support System) to independently control and manage a diverse range of health applications. The master Patient Agent depicted in figure 4.7(b), playing the role of a broker or controller, determines the resource requirements for supporting a particular healthcare service. Figure 4.7(b) depicts that the master Patient Agent comprises two functional components, namely, a slice manager and a slave manager. The slave manager creates a slave Patient Agent dedicated to serve a specific health service. The slice manager asks TC via the OSS to allocate a network slice to run the newly created slave Patient Agent. For instance, the slave manager can assign a slave Patient Agent to run the Blockchain algorithms, including consensus mechanism to a network slice supporting higher computing because Blockchain tech-

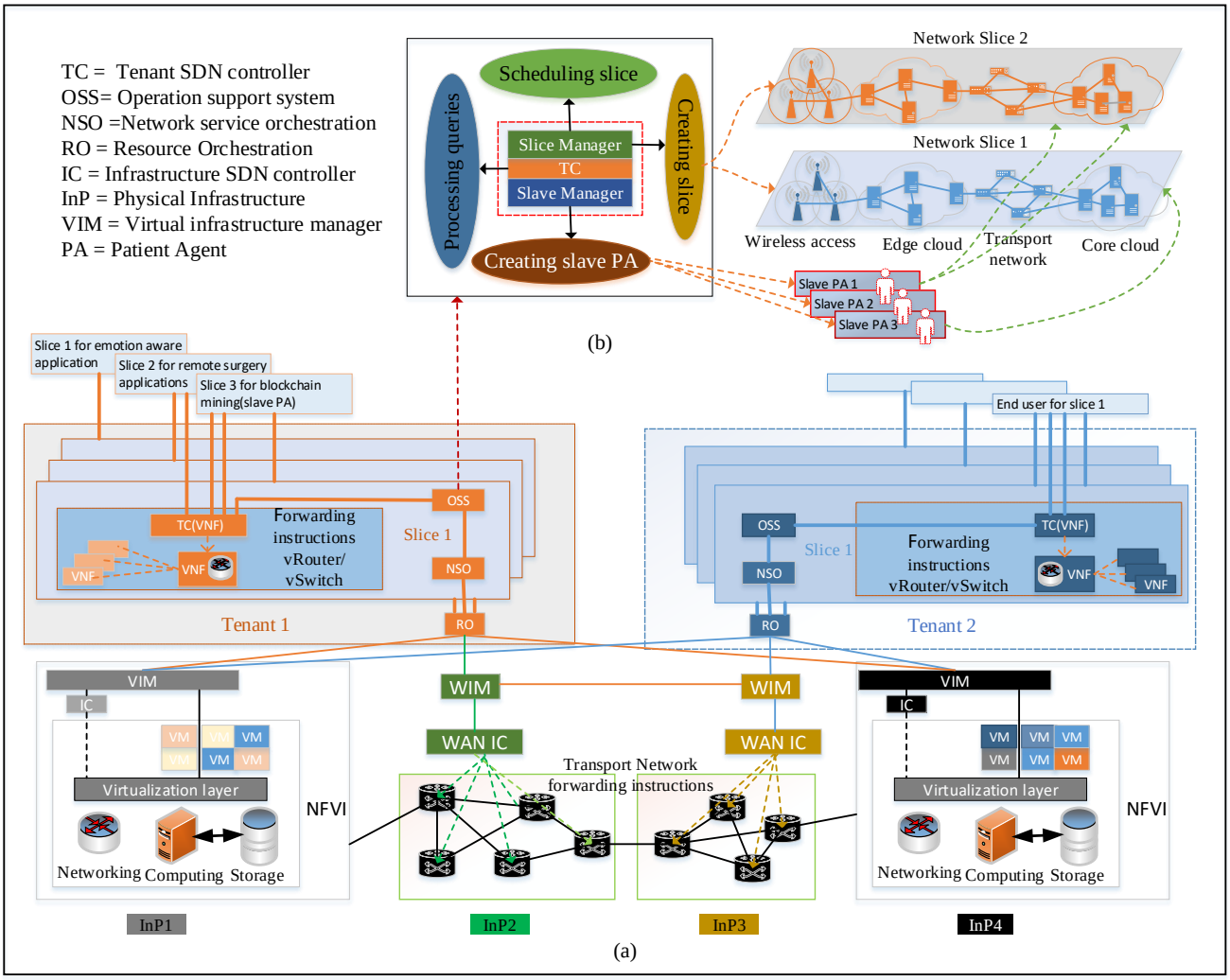


Figure 4.7: The Patient Agent on ETSI 5G architecture

nologies require incredibly high computing resources for processing health data in near real-time. Similarly, the slave manager assigns another salve Patient Agent for migrating a task to a network slice with ultra-low speed. A slave Patient Agent for managing remote heart surgery runs on a network slice that has ultra-high bandwidth, very low latency and reliable communication channels. Multiple instances of a slave Patient Agent run on mobile access, Edge and core Cloud of a network slice to make the system fault-tolerant.

4.3.3 Functionalities of the Patient Agent

The Patient Agent supports many health operations including task migration, storage and security management, access control, task execution and Blockchain management. In this framework, some of these significant operations depicted in figure 7.4 are described below.

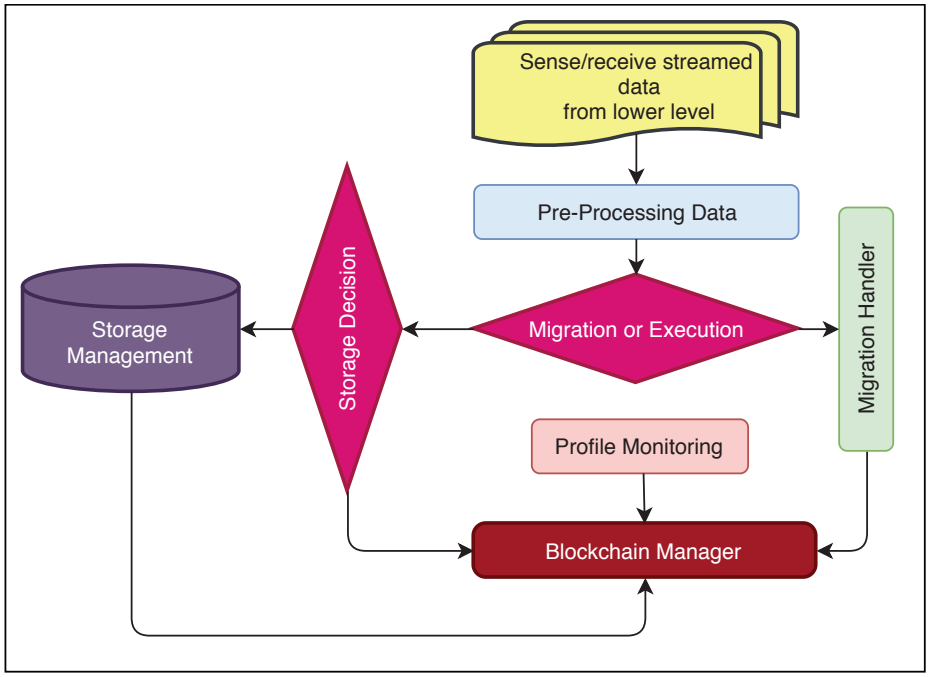


Figure 4.8: The functionalities of the Patient Agent at three levels

4.3.3.1 Migration Handler(MH)

IoT devices are incapable of providing considerable communication and computation resources required to process a task with low latency. The capabilities of medical IoT devices in terms of processing and storage are less than that of an average computer. Medical IoT devices can save a significant amount of power and make processing faster by offloading heavyweight tasks to their embedded Edge devices[83]. Task offloading in Mobile Cloud Computing(MCC) has been extensively investigated in state-of-the-art research. But, there is still a room to design a secure Blockchain leveraged task offloading mechanism in MEC(Mobile Edge Computing) for the healthcare domain. Maintaining patient's privacy and security during migrating tasks to other remote Patient Agents is indispensable to meet services QoS.

MAUI[438] and CloneCloud[439] determined optimal location(local or remote machine) to execute a task using an offline linear optimization method. They assumed that the energy or time

required to offload a task to a machine is known to the system. In this case, the local device needs to collect QoS related information from the remote machine. A remote machine might lie to a local machine, and there needs a secure mechanism to obtain processing capabilities related information of the remote machine. Further, the decision of offloading a task by solving a linear optimization problem involves high computational cost. Further, they statistically partitioned the task offline, which is thought to be non-optimal.

ThinkAir[440] and CADA[441] brought a modification of MAUI's algorithm by considering average execution time based cost required in local device and remote device. They both devised their offloading algorithm for static execution environments, and they assumed these parameters remain unchanged regardless of location or time. Consequently, the proposed methods did not perform well in a dynamic execution environment. Khoda[442] applied non-linear optimization Lagrange multiplier to decide code offloading. They could improve energy consumption because non-linear Lagrange multiplier's operation is lighter than that of the linear optimization method. Linear regression was used to predict the execution time of the offloading task. In healthcare, execution time depends on the data size of the task that varies over time. So, the linear regression-based prediction did not estimate the accurate execution time.

We propose to use a Blockchain to store execution environment parameters of a remote machine. The Blockchain's nodes authorize environment parameters of the potential remote devices that want to take part in executing migrated tasks. The local machine retrieves these parameters from the Blockchain to outsource tasks to neighbouring or remote devices. Further, offloading tasks are categorized based on privacy and time sensitivity to preserve patient privacy and meet QoS, respectively. The Hungarian assignment method that is solvable in polynomial time discovers a set of optimal remote devices for a set of tasks.

4.3.3.1.1 Task Migration Algorithm: Tasks performed on health data require various levels of resource requirements and security, depending on their complexities, data sensitivity and size. For instance, health data filtration, fusion, compression, and other data mining analyses often require high computing power and massive storage available only on Edge devices or Cloud servers. An early warning module should be executed in Edge devices while streaming data from IoT devices without causing much communication delays that are required to transfer the task to the Cloud server. We consider the facts mentioned above to devise an offloading algorithm.

Figure 4.9 presents the offloading mechanism of **MH**. A Fog Agent can outsource its tasks to a neighbouring or remote Fog Agent that have spare capacity. In general, a local machine allots a task to a remote machine if any of the following condition is true:

- Condition 1: If energy consumption to execute the task in the local machine is higher than the energy consumption to transmit the task to the remote machine.
- Condition 2: If processing time of the task in the local machine is higher than the processing time of the task in the remote machine.
- Condition 3: Condition 1 and 2

Our task migration algorithm depicted in figure 4.9 is described as follows. First, tasks are classified before deciding to outsource them. Secondly, two separate matrices are created for each group of tasks. One matrix contains the amount of energy consumption required for a local machine to transmit tasks to the available remote machines. These tasks can be tolerant to some degree of delay without compromising QoS. Another matrix contains response time required to

complete the processing of tasks in available remote machines. These tasks are delay-sensitive and need ultra-low latency. Thirdly, privacy-sensitive tasks are scheduled among homogeneous Patient Agents(the agent instances of the same patient) and other kinds of tasks might be scheduled among any Patient Agents regardless of homogeneous or heterogeneous. In both cases, the Hungarian assignment algorithm runs if the number of tasks is $n \geq 2$.

It is assumed that a Patient Agent on the smartphone or Fog has multiple tasks($j = 1, 2, \ldots, m$) to be assigned to multiple neighbouring or remote agents($i = 1, 2, \ldots, n$).

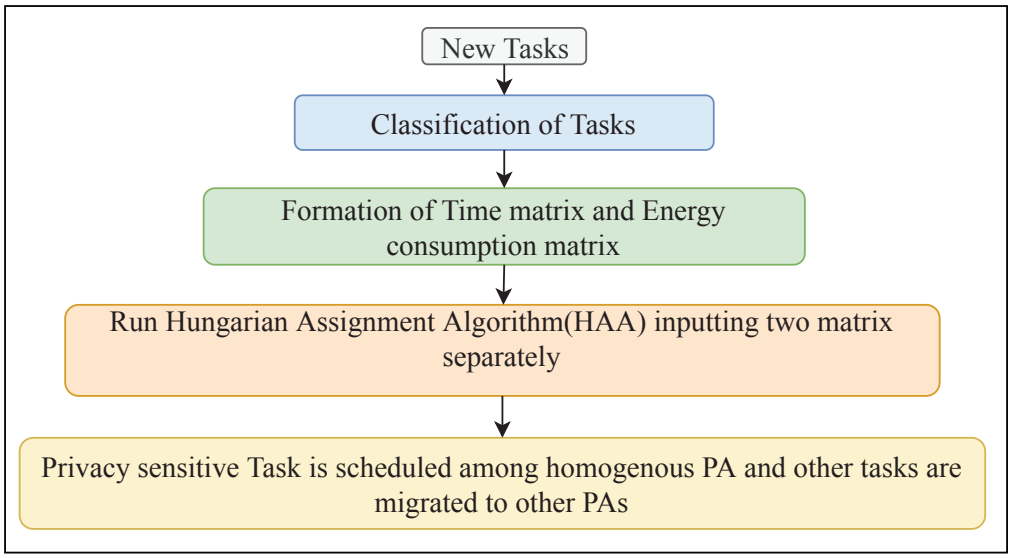


Figure 4.9: The privacy aware task migration process

- **Task Classification:** Each task on health data is classified as Privacy Sensitive Task(PST) or Normal Task(NT) following the method described in section 4.3.3.1.2. Security-related tasks including encryption/decryption, key generation, tasks pertaining to data a patient considers sensitive. Tasks on low sensitive or not sensitive data, may include ECG, body temperature commonly regarded as normal tasks. Both PST and NT are further classified as Delay Sensitive Task(DST)(heart surgery, emotion aware) or Delay Tolerant Task(DTT) respectively.

- **Time Based Cost Matrix Formation:**

  This matrix contains execution time required for all the DSTs(Delay Sensitive Task) to be executed in m number of available remote Agents. If the execution time of a DST in a remote machine is higher than that of execution time in local machine, a large number is inserted into the corresponding place of that task and agent in the matrix. Consequently, the agent is not chosen for executing that task. The execution time of a task in the local Agent is calculated as follows;

  If the local Agent's MIPS(million instructions per second) is μ_l and a task(j) has I number of instructions including its execution environment, the time needed for the local Agent to complete the task(j) is estimated as follows:

  $$\text{Response Time} = \text{Execution Time} + \text{Queue Latency}$$

  $$T_{j,l} = \frac{I_j}{\mu_l} + \sum_{j=1}^{k} \tau_{j,l}$$

where the queue latency of the local Agent is $\sum_{j=1}^{k} \tau_{j,l}$. This indicates that the Agent needs to execute k number of pending tasks besides the current one. $\tau_{j,l}$ indicates the execution time of a task(j) that is waiting in the queue of the local Agent.

The aim is to schedule tasks to different remote Agents. The response time from a remote Agent does not only depend on the processing capabilities of the Agent but also the quality of communication link between the local and remote Agents. The response time for a task from a remote Agent is the summation of the propagation time, transmission time, queue delay and execution time required for the remote Agent. The summation of propagation and transmission time is the uploading time. The propagation time is the time for one bit to travel from one router/switch to the next router/switch, and it depends on the distance between two entities and the speed of the communication medium. Transmission time represents the time to get out all the bits of a task from a device to the transmission wire. The response time of a task(j) from a remote Agent is estimated as follows:

$$\text{ResponseTime} = \text{Transmission Time} + \text{Propagation Time} + \text{Execution Time} + \text{Queue Latency}$$

$$T_{j,f} = \frac{\omega_j}{\beta_{l,f}} + \frac{d_{l,f}}{\nu} + \frac{I_j}{\mu_f} + \sum_{j=1}^{k} \tau_{j,f}$$

where the amount of data involved in the task(j) is ω_j, $d_{l,f}$ is the geographical distance between the local and remote Agent, ν indicates propagation speed of the link between the two Agents and μ_f represents CPU speed of the remote Agent in MIPS. $\beta_{l,f}$ is the bandwidth of the communication link between the local and remote Agent. The following matrix for m number of tasks and n number of available remote Agents is formed to input it in the Hungarian Assignment Algorithm that discovers optimal allocation of m number of tasks to n number of remote Agents in terms of execution time.

$$\begin{Bmatrix} T_{1,1} & T_{1,2} & \dots \\ \dots & \dots & \dots \\ \dots & \dots & T_{m,n} \end{Bmatrix}$$

$T_{1,1}, T_{1,2}, \dots, T_{1,n}$ represent execution time of task 1 in remote Agent 1 and 2 and so on. Similarly, $T_{2,1}, T_{2,2}, \dots, T_{2,m}$ represent execution time of task 2 in remote Agent 1 and 2 and so on.

Here,

$$T_{ij} = \begin{cases} T_{j,f} & \text{if } T_{j,l} > T_{j,f} \\ \infty & \text{if } T_{j,l} < T_{j,f} \end{cases}$$

- **Energy Based Cost Matrix Formation:** In case of Delay Tolerant Tasks(DTTs), the local Agent can attempt to save energy consumption by assigning tasks to other neighbouring or remote Agents.

  The energy that the local Agent consumes to execute a task (j) is estimated as follows:

  $E_l = \rho_x \times \frac{I_j}{\mu_l}$. Where ρ_x is the power consumption rate of the local Agent while executing a task. I_j is a number of instructions in the task.

The local Agent consumes energy to transfer a task to a neighbouring or remote Agent. The energy consumption while offloading a task is occurred due to network interfaces and spending idle time of the local Agent(if the agent does not have any tasks in the queue). The energy that the local Agent consumes to offload a task can be estimated as follows:

$E_f = \rho_d \times Tj, f + \varepsilon_{trans}$

Where ρ_d is a power consumption rate of the local Agent during idle mode. $T_{j,f}$ is the response time of the task(j) if it is assigned to a remote Agent. The energy consumption due to network interface while transmitting the task can be estimated as follows:

$\varepsilon_{trans} = \rho_l \times \frac{\omega_j}{\beta_{l,f}}$

where ρ_l is the power consumption rate of the local Agent. The following matrix contains energy consumption of the local Agent for m number of tasks if those tasks are assigned to n number of remote Agents.

$$\begin{Bmatrix} E_{1,1} & E_{1,2} & \dots \\ \dots & \dots & \dots \\ \dots & \dots & E_{m,n} \end{Bmatrix}$$

$E_{1,1}, E_{1,2}, \dots, E_{1,n}$ represent energy consumption of local Agent to execute task 1 in remote Agent 1 and 2 and so on. Similarly, $E_{2,1}, E_{2,2}, \dots, E_{2,n}$ represents energy consumption of local Agent to execute task 2 in remote Agent 1 and 2 and so on.

Here,

$$E_{ij} = \begin{cases} E_{j,f} & \text{if } E_{j,l} > E_{j,f} \\ \infty & \text{if } E_{j,l} < E_{j,f} \end{cases}$$

Representation of Hungarian Assignment: The Hungarian Algorithm is separately run inputting two matrices for two genres of tasks. The mathematical presentation of the Hungarian Assignment on the basis of a time-based cost matrix is shown in equation (4.1).

$$\begin{aligned} \min_{t,\,x} \quad & \sum_{i=1}^{n} \sum_{j=1}^{m} t_{ij} x_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{m} x_{ij} = 1, \quad j = 1, \dots, m, \\ & \sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, \dots, n \end{aligned} \tag{4.1}$$

where

$$x_{ij} = \begin{cases} 1 & \text{if the device is assigned } j^{th} \text{ task} \\ 0 & \text{if the } i^{th} \text{ device is not assigned } j^{th} \text{ task} \end{cases}$$

4.3.3.1.2 Identification of Sensitive Tasks: State-of-the-art research[443–446] has already addressed the issue of identifying sensitive medical records. Yang[443] presented a model to identify protected health information from the clinical records. The identification method involves the machine learning approach with keyword-based and rule-based techniques to separate protective

health terms. Jindal et al. [444] presented semi-supervised techniques to detect sensitive words from clinical narratives. They used the SNOMED CT ontology of health concepts to train the model, which eliminates the needs of an annotated and unannotated dataset. Authors also applied a rule-based method to classify the negotiation words and experiencer. Sanchez[445] proposed an automatic sensitive terms detection system using an information-theoretic approach. They used the web-based corpus to make the solution more generalized and domain-independent. Tesfay[446] presented an architecture for assessing user-centred privacy risk. The architecture includes three components: privacy detection system(PDS), risk communication manager(RCM), and privacy quantification(PQ). RCM determines the privacy level of personal data like high, medium and low sensitive. PQ quantifies the privacy risks based on probabilistic and combinational techniques.

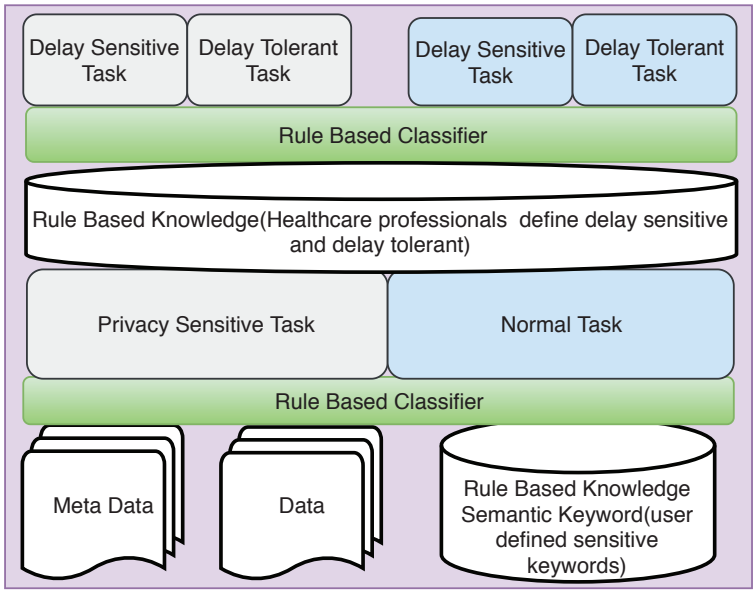


Figure 4.10: The framework for detecting the sensitivity of a task

Here, we used a sensitive content identification framework designed following[443–445]. The framework automatically categorizes sensitive health data for MH(Migration Handler) because tasks on the sensitive data is considered to be sensitive. The framework depicted in Figure 6.6 has two kinds of rule-based classifier. The data and metadata containing task information are fed into the first rule-based classifier. The classifier applies the rules stored in the Rule Base Knowledge(RBK). The RBK stores the user's feedback and expert's feedback regarding the sensitive keywords in health data. The first rule-based classifier categorizes the task as Privacy Sensitive Task(PST) or Normal Task(NT) using RBK. Next, the task(PST or NT) is classified by the second rule-based classifier as Delay Sensitive(DST) or Delay Tolerant Tasks(DTT). The second classifier applies the rules from another rule-based knowledge formed with the feedback collected from the healthcare professionals.

4.3.3.2 Profile Monitoring(PM)

The Migration Handler of a local Agent aims to outsource and distribute computing tasks to neighbouring and remote Patient Agents that have additional computing resources. Each Patient Agent, therefore, needs to know the profile of other Agents on the peer-to-peer Edge network. The Profile Monitoring(PM) module requires queue latency, CPU speed, availability, and bandwidth informa-

tion of the available remote Agents to pass on to the Migration Handler. The malicious Agents might lie to a client Agent about their performance parameters. To address this issue, we propose to utilize Blockchain to manage and store performance parameters for migrating tasks. Every Patient Agent registers their performance parameters on the Blockchain.

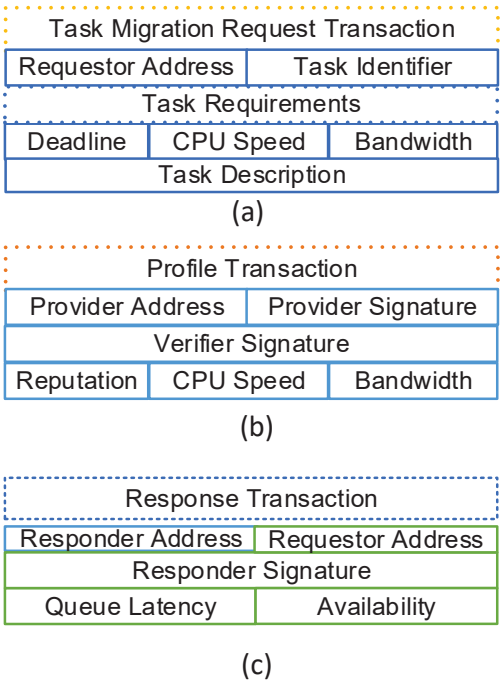


Figure 4.11: The transactions for migrating tasks

The PM creates several transactions (depicted in figure 4.11) related to task migration. For instance, a Patient Agent issues a Profile Transaction(PT) containing CPU processing, storage capacity and other parameters including network capacity when it first joins in the Blockchain. Miner nodes on the Blockchain verify these parameters packed in the transaction. A Patient Agent might include additional resource capabilities over time. If it does so, it needs to make a new Profile Transaction in the Blockchain. In this case, Blockchain nodes also process, and validate these transactions. The PM of a PA broadcasts a Task Migration Request Transaction (shown in figure 4.11(a)) throughout the peer-to-peer Blockchain network when it needs to outsource tasks. The other Agents with available resources reply to the PM by making a Response Transaction (shown in figure 4.11(c)) that contains the dynamic resource information such as queue latency of the remote Agent. The local Agent can retrieve static profile information of that Agent from the Blockchain.

4.3.3.3 Execution Unit(EU)

The execution unit(EU) of the PA performs the processing of health data. The processing might include filtration, fusion, generating a warning, automatic diagnosis and other operations. A Patient Agent has the option to choose an EU among its own EU, other Patient Agent's EU and smart contract-based EU. A smart contract refers to a set of rules encoded using a specific programming language[447]. Every Blockchain node stores coded rules for a smart contract. A Smart Contract is triggered when a transaction specified to that smart contract is issued in the Blockchain network.

The Patient Agent might make different kinds of smart contracts for processing a patient's health data. The followings are a few examples of the smart contract.

- **Smart Contract for Registration:** This contract is executed once a Patient Agent registers in the Blockchain for the first time.
- **Smart Contract for Data Filtration:** The contract contains code for clinically uninteresting health data filtered out.
- **Smart Contract for Data Classification:** The contract holds the procedure for categorizing health data as normal and abnormal. The contract is triggered upon the request of data classification.
- **Smart Contract for Warning Generation:** The contract holds the code to generate alarm after analyzing continuously streamed medical data.
- **Smart Contract for Task Migration:** This contract is triggered while migrating tasks to high computing devices.

4.3.3.4 Storage Determination(SD)

Health-related data can be stored on diverse repositories including government-managed repositories (e.g. myGov electronic health record in Australia), Blockchain, on healthcare service provider servers, on private Cloud servers, on a patient's personal computer or any other devices. Different repositories provide a different level of security and privacy. Patients have diverse privacy preferences. The SD will model a patient's privacy preferences and experts knowledge of security to automate decisions regarding preferred storage for health data. This module will determine the storage repositories for data stream rapidly from wearable sensors and other kinds of health data.

4.3.3.5 Blockchain Manager(BM)

Blockchain's structure offers a couple of advantages such as tamper-proof storage, patient's privacy and processing data without the need to trust third parties. The Blockchain maintains a unique ledger replicated amongst multiple nodes. The ledger is formed with a series of confirmed Blocks connected between them in linked list fashion where Block comprises a certain number of transactions packed in a secure Merkle tree. Blockchain nodes add a new Block in the current ledger by running a consensus mechanism.

The BM can apply Blockchain technologies to perform task migration, store streamed health data, operate diagnosis, and control access. The BM also undertakes security services such as key management, encryption/decryption of health data, make various kinds of health data transactions to be inserted into the Blockchain, and participates in consensus protocol that is required to add a new data Block in the Blockchain. In this section, we describe a lightweight modified Proof of Stake consensus mechanism for our IoT healthcare architecture.

The components of a standard Blockchain(used in Bitcoin) are illustrated in figure 6.7 before discussing our customized Blockchain.

1. The Blockchain operates on a peer-to-peer network depicted in figure 6.7 (a). Nodes of this network is categorized into three groups: half nodes, general nodes and Miner nodes.

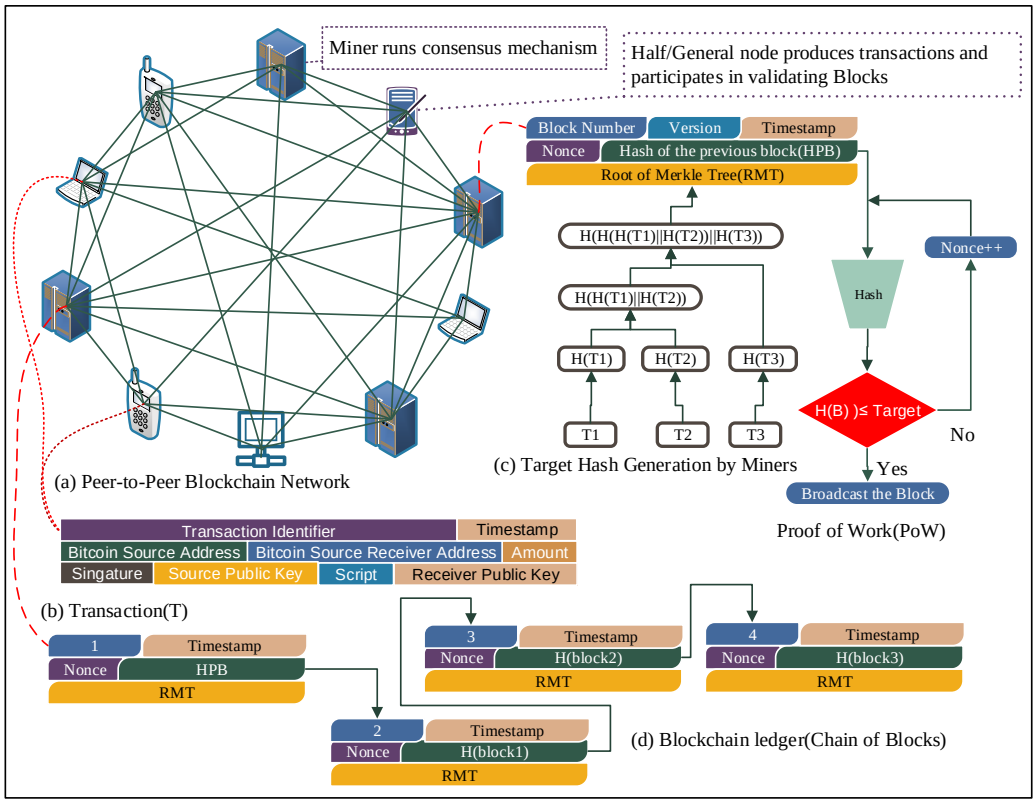


Figure 4.12: The Bitcoin Blockchain

2. The half node/public node produces transactions formatted as in figure 6.7(b) and broadcast throughout the peer-to-peer network.

3. The Miner nodes collect a certain number of transactions and pack them into a Merkle tree to create a Block. Each Miner repeatedly inputs the Block into a cryptographic hash function incrementing the nonce field by one every time as long as the hash function produces a target hash code for the Block. This process depicted in figure 6.7(c) is called Proof of Work(PoW). Only one Miner which first publishes the Block with Proof of Work receives the reward for doing this.

4. Finally, all nodes except the half-node link the Block to the end of the existing chain as depicted in Figure 6.7(d) by running a verification process.

4.3.3.5.1 The Lightweight Consensus Mechanism: In our eHealth architecture, the smartphone transmits its captured health data to NEAR processing devices(Fog devices). The NEAR processing level should execute the Blockchain operations to process medical IoT data in near real-time. But, Blockchain technologies requires high computational cost and storage, and NEAR processing devices do not have appropriate capabilities to accommodate a Blockchain. However, Blockchain technologies can be adapted in an IoT healthcare if Blockchain operations are partitioned. We can allot Blockchain operations to three layers of the IoT healthcare architecture, considering devices' capabilities of these layers. For example, the smartphone Agent in the sensing layer can define the structure of transactions and initiates data flow. The Fog Agents in the NEAR processing layer can execute a lightweight consensus mechanism and store metadata about

a Block. This reduces delay in Block's confirmation on the Blockchain because NEAR processing devices are located at one hop from the medical sensors. The Agents in the FAR processing layer can permanently provide storage for the Blockchain-based health ledger. The Cloud Agents insert Blocks into Cloud ledger after the Fog Agent confirms the Block by running a consensus mechanism. The PoW, pBFT, PoS and DPoS consensus mechanism has the following downsides that prevented their use:

The Proof of Work(PoW) used in Bitcoin and Ethereum is the most decentralized consensus method. PoW is employed to solve the byzantine general's problem. In PoW, the Miners write a new Block in the existing chain of Blocks through generating a target hash of the new Block. The Miners compete to come up with the target hash, and the winner obtains a certain amount of reward. Every node ensures containing an identical version of the newly constructed Block in the Blockchain through a validation process. The PoW protects the Blockchain from DoS, and Sybil attacks as attackers are not being encouraged to invest a tremendous amount of computational resources required for the PoW. But, this consensus mechanism is not suitable for processing streamed medical IoT data because the medical data transactions necessitate faster processing to meet patient's QoS. The Bitcoin network generally requires approximately 10 minutes [405] to reach on the agreement for a Block due to using PoW mechanism.

pBFT (Practical Byzantine Fault Tolerance) discussed in the first chapter shows better performance for enterprise consortium Blockchains. The participants in this consensus mechanism are partially trusted. However, the downside of the pBFT is that the number of messages exponentially increase ($O(n^k)$, where n is the messages and k is the number of nodes) due to multicast nature of the protocol[448]. The pBFT consensus model is only effective when the distributed network has a small number of nodes[449]. The pBFT is vulnerable to Sybil Attacks where one entity maintains several IDs. Malicious nodes find it more difficult to launch Sybil attacks as the number of nodes in the network grows, however pBFT has limited scalability and is inefficient in big networks[450].

One of the most proficient consensus methods is Proof of Stake (PoS)[405] in terms of scalability and processing time. In this process, prospective Miners have to lock their coin to the system. A Miner having higher share makes the next Block, and it receives incentives for this. The PoS consensus mechanism has risk in price volatility that results in the loss of user's wallet[451]. Many cryptocurrencies that utilize PoS tend to set limits on the number of coins, a potential miner requires. This promotes participants to unite in "pools" so that mining rewards can be proportionately divided among the "pools". As the majority of assets are held by a few participants, the risk of centralization is significant, affecting newer cryptocurrencies that offers low price for trading[452]. There might be a chance of a drop in cryptocurrency turnover because users strive to hold currencies on their accounts for as long as possible in order to maximize profits. There is also a provision of trusting third party services in PoS[453].

Delegated Proof of Stake(DPoS) is a variation of standard PoS. With DPoS, users or a group of delegates select a set of witness nodes through a voting mechanism[454]. The weight of a delegate's vote for a witness node is proportional to the amount of the witness node's deposited coin in the system. A witness obtaining the maximum number of vote from the panel accumulates the transactions and organizes those into a Block. Other witness nodes verify the newly created Block to confirm it in the Blockchain. In DPoS, decentralisation is difficult to achieve because only a small number of holders participate in making decisions on the Blockchain, allowing for censorship and conspiracy. Critics claim that DPoS sacrifices decentralization for scalability[455]. Low numbers of participates in voting can lead to centralization, which means power could be concentrated in the hands of a small number of token holders. A 51% attack is easier to plan because fewer individuals are in charge of keeping the network alive[456]. The DPoS blockchain

is vulnerable to weighted voting issues. Users with a minor stake can opt out of voting if they believe their vote is insignificant. Delegators must be well informed and appoint honest witnesses to execute DPoS protocol and make decisions effectively.

To summarize, some drawbacks of this mechanism are: few nodes can dominate the entire network, which makes the system vulnerable to a 51% attack. Nodes having a high stake can influence the Blockchain network more than that of nodes with a small stake due to assigning the weight of a vote based on the stake a node holds. Further, delegates might collude to vote for a particular group of witness nodes.

We propose a modified PoS to mitigate some of the standard PoS and DPoS mechanisms' drawbacks mentioned above. The section below describes the modified consensus mechanism demonstrated in Algorithm 6.

- **Cluster Formation:** In the NEAR processing layer, Fog/Edge Agents within a certain geographical range(R) form a cluster. The Figure 7.2 depicts three clusters: C_1, C_2, and C_3 at the NEAR processing level. Every cluster has a random number of Patient Agents, where one member is elected as cluster head. The cluster head(also called leader) participates in running the consensus protocol of the Blockchain by locking a certain amount of stake to the system. Depositing a certain amount of digital coin is mandatory for every Miner. The node identified as malicious one loses its stake. The node also receives a negative review from peer Miners so that it has a slim chance to be a Miner next time.

- **Leader (Cluster Head) Nomination:** A cluster head(CH) is selected from each cluster considering multi-criteria of the member nodes. The selection criteria include a node's performance parameters, reputation, and the amount of stake. These criteria are combined using a Fuzzy Inference System(FIS) to estimate a fitness value. Every node's information regarding the mentioned criteria is recorded in the Blockchain and can be retrieved from the Blockchain when they are needed.

  The performance parameters include the processing speed of an Agent's device, memory capacity, availability, distance coefficient of variation, and transmission delay. Here, processing capabilities in MIPS, memory capacity and availability are symbolized as p_1, p_2, and P_3 respectively. These parameters are normalized in the range from [0 to 1] and then those normalized values are summed up as follows:

  $$\gamma = \sum_{p=1}^{3} \frac{1}{(1 + e^{-p_i})}$$

  A node whose distance from the other nodes in a cluster is uniform should be selected as a cluster head. The reason is that a cluster head which is closer to most of the member nodes and a bit far away from few other member nodes might experience an inconsistent delay to receive/send data from/to all the member nodes. A node with almost uniform distance from other nodes is appropriate as a cluster head for synchronizing timestamp between nodes. Distance coefficient of variation(CoV) can be applied to estimate the consistency of a node with respect to distance from other nodes in a cluster. If $d_1, d_2, \ldots, d_n$ are distance of other nodes from a node i, CoV_i is calculated as follows. CoV is the result of standard deviation divided by the mean of a set of values.

If the mean $\mu_i = \frac{\sum_{k=1}^{n} d_j^k}{n}$ and standard deviation $\sigma_i = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(d_j^k - \mu)}$ then Coefficient of Variation $CoV_i = \frac{\sigma_i}{\mu_i}$

Next, delay refers to the time required for one bit to send from the source to destination. Here, the harmonic average delay concerning a node(i) refers to a harmonic average of the amount of delay to receive one bit from other nodes in the cluster.

$H_i(t) = \frac{n}{\sum_{j=1}^{n}\frac{1}{t_j}}$, where, j = 1 to n member nodes need $t_1, t_2, ..., t_n$ respectively to send one bit data to a particular node i.

The harmonic average delay is normalized in the range [0 to 1] as follows.

$$\tau_i = \frac{1}{1 + e^{-\frac{1}{H_i(t)}}}$$

The higher CPU speed, memory capacity, availability of a node and the lower average delay and coefficient of variation with a Miner, the better the Miner is. Therefore, each node in the cluster combines its performance as follows.

$$P_i = \frac{1}{(1 + e^{-\frac{\gamma}{\tau \times CoV_i}})}.$$

The second criteria for selecting a cluster is reputation. An Agent receives a transaction containing a positive or negative reputation in the range of [1 to 5] when the Agent serves a requestor's service. A requestor can pick a value from this range [1 to 5] as a positive or negative reputation on the basis of QoS including timely service, accuracy and others an Agent offers. If an Agent serves multiple services from multiple requestors, it obtains a reputation transaction for every service it offers. Blockchain records each reputation an Agent receives. The positive or negative reputation that a requestor provides to an Agent is multiplied by the requestor's positive reputation and divided by the negative reputation of the service providing Agent. The initial positive or negative reputation of each Agent is assumed 1.

It is supposed that an Agent(i) already obtained positive reputation from n number of service requestors $\alpha_1, \alpha_2, \ldots, \alpha_n$ and negative reputation $\beta_1, \beta_2, \ldots, \beta_n$. The positive reputation of these service requestors is $\omega_1, \omega_2, \ldots, \omega_n$ The ultimate reputation for a service providing Agent while nominating cluster head is estimated as follows. $r_i = \sum_{j=1}^{n} \frac{\alpha_j \times \omega_j}{\beta_j}$

The aggregated normalized reputation is as follows: $R_i = \frac{1}{1+e^{-r}}$.

The third criteria named digital coin(c) that an Agent(i) has in the system is normalized as follows: $S_i = \frac{1}{(1 + e^{-c})}$.

Now, every Agent in a cluster calculates their fitness(f_i) using the criteria outlined above. A Fuzzy Inference System(FIS) is used for this because fuzzy rules can represent sophisticated heuristics more appropriately than crisp rules. The input of the FIS is an Agent's

performance, reputation and the current stake. An Agent with higher fitness is delayed for a short period before declaring itself cluster head. The member node in a cluster waits for the following period presented in equation (4.2)

$$T_i = \Delta T \times (1 - \frac{f_i}{\sum_i^n f_i}) \pm \lambda \tag{4.2}$$

Where ΔT is the time interval for electing cluster head, and λ represents a short random time duration that is used to differentiate waiting time for the Agent having the same fitness. The Agent that expires its waiting time broadcasts its identifier throughout the cluster. The other cluster members verify the estimated fitness of the Agent and acknowledge their approvals for this Agent. The Agent finally wins as a leader of the cluster only if it obtains specific numbers of approvals. Once a leader makes a Block, a specific value is deducted from its original fitness so that the chance of other node's being leader increases next time.

- **Super Leader Nomination:** A super Leader is randomly selected from the selected set of cluster heads. This super leader is responsible for making new Block packing a certain number of transactions from the transaction pool. A new super leader is elected after the current super leader prepares a certain number of Blocks. A new round begins when every cluster head eventually becomes super leader. Every cluster head takes part in verifying a new Block before broadcasting it throughout the network. The Fog Agent stores the metadata about a Block and transmits the complete Block to FAR processing layer for the permanent storage. Half of the reward for mining is awarded to the super leader, and the rest of the half reward is equally divided among the cluster heads.

  The super leader selection is as follows: each cluster head except the Agent that was already elected as super leader generates a random number between 0 and 1 according to equation (4.3). A cluster head becomes a super leader if its random number is less than the threshold stated in equation (4.3) and obtains the threshold number of votes from the other cluster heads.

$$I = \begin{cases} \frac{p}{1-p[r\bmod(1/p)]}, & N \in G \\ 0, & \text{otherwise} \end{cases} \tag{4.3}$$

  where p is the percentage of cluster head in the Fog network, N is a total number of cluster head, the r is the number of rounds of selection, and G is the set of cluster heads that have been elected as super leader in round $\frac{1}{p}$.

Every node in a cluster can participate in the proposed PoS by turns. This consensus mechanism is less vulnerable to 51% attack than DPoS as a leader comes from each cluster. Unlike PoS, the rich node has less chance to be a leader of a cluster because the cluster head is not only selected based on the locked coin but also reputation and performance parameters. Further, the reduction of specific points from the current leader's fitness prevents the node from being a leader for the subsequent round.

Algorithm 6: Modified PoS Consensus Protocol

Data: Performance Transaction(PT), Reputation Transaction(RP), Stake Transaction(ST), Agent number(n_k) in a cluster

Result: a set of healthy Miners

Every Fog Agent generates PT_i, and ST_i for themselves and obtains RP_j from service providers (j).
Form clusters with Fog nodes within a threshold range(R).
for *each cluster* k = 1 *to* l **do**
    while leaderElection = true **do**
        for *each member Agent* i = 1 *to* n_k *of a cluster* **do**
            Extract parameters from PT_i, LST_i, RP_i to produce P_i, , R_i , and S_i
            $P_i = \frac{1}{(1 + e^{-\frac{\gamma}{\tau \times CoV_i}})}$.
            $r_i = \sum_{j=1}^{n} \frac{\alpha_j \times \omega_j}{\beta_j}$
            $R_i = \frac{1}{1+e^{-r}}$
            $S_i = \frac{1}{(1 + e^{-c})}$
            $f_i \leftarrow$ **fuzzyInferenceProcess**(P_i, R_i, S_i);
        end
        $T_i = \Delta T \times (1 - \frac{f_i}{\sum_i f_i}) \pm \lambda$
        Every member node in the cluster(j) sets their timer(T_i).
        if T_i is expired **then**
            Broadcast nodeId throughout the cluster for approval.
        end
        if approvalCount[nodeId] $\geq \frac{2}{3} \times n_k$ **then**
            $leader_j \leftarrow$ nodeId;
            $f_{nodeId} \leftarrow f_{nodeId} - \varepsilon$
            leaderElection $\leftarrow$ false
        end
    end
end
superLeader $\leftarrow$ **selectSuperLeader**($leader_1, ..., leader_m$)

4.3.3.5.2 Fuzzy Inference System(FIS) to assess a node's fitness A fuzzy expert system is a collection of fuzzy rules and membership functions that are used to reason about data. Fuzzy inference process that refers to a process of mapping a given input to n output by using the theory of Fuzzy sets. The Fuzzy inference process involves four steps: Fuzzification, rule evaluation, aggregation, and defuzzification. The functional blocks of the FIS depicted in Figure 7.5 to generate fitness used in the consensus algorithm is briefly described below:

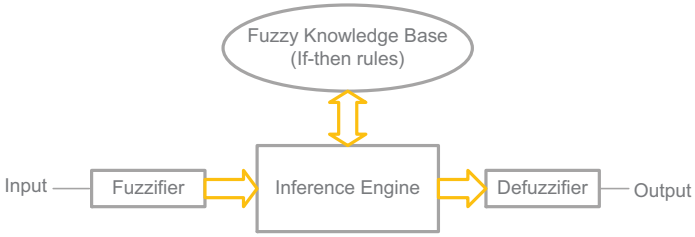


Figure 4.13: The FIS for determining node's rate

- **Fuzzifier:** The first step of fuzzy inference is to map crisp(numerical) inputs into degrees to which these inputs belong to respective fuzzy sets. Fuzzifier converts crisp inputs to linguistic variables applying membership functions such as triangular, trapezoid or Gaussian functions. Figure 7.7 shows the conversion of crisp input(performance) using MATLAB FIS. The numerical value in the range of [0 to 1] is expressed in a linguistic variable: low, medium and high.

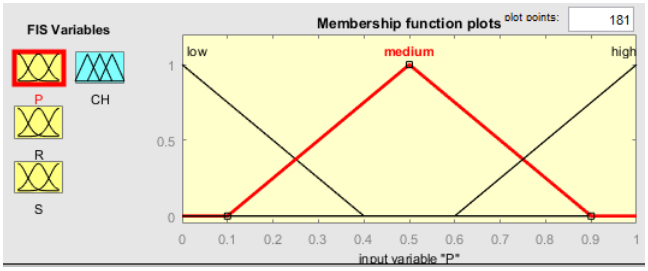


Figure 4.14: The membership function for performance parameter

- **Inference Engine(Rule evaluation and aggregation):** Inference engine stores fuzzy rules. These rules consist of "If" and "then" statement. This step takes the fuzzified inputs and applies them to the antecedents of the fuzzy rules. Few "If-then" rules for evaluating fitness are represented in Table 4.3. In Figure 7.3, **P**, **R** and **S** stand for miner's performance, reputation score and the amount of stake receptively.

- **Defuzzifier:** Finally, Defuzzifier converts the fuzzy outputs generated by Inference Engine to a single crisp number using Centre of Gravity (COG) or other methods such as the centre of Area(AOC), bisector of area(BOA). A fuzzy output is generated from each "ifthen" rule firing. Each output fuzzy is the input of the Defuzzier. The Defuzzier aggregates fuzzy output set into a crisp out as depicted in Figure 7.3 where multiple rules have been fired. Figure 7.3 illustrates that if P(performance score) = 0.476, R(reputation score) = 0.608 and S(score from deposited coin) = 0.331, the aggregated score using equation (4.4) is 0.805(indicates the probability of the node being leader).

Table 4.3: The ifthen rules for the FIS

SL. No.	Rules
1	**if**(P is *low*) **and** (R is *low*) **and** (S is *low*) **then** (CH is *low*)
2	**if**(P is *low*) **and** (R is *medium*)**and** (S is *low*) **then** (CH is *low*)
3	**if**(P is *low*) **and** (R is *high*)**and** (S is *medium*) **then** (CH is *medium*)
4	**if**(P is *medium*) **and** (R is *low*)**and** (S is *medium*) **then** (CH is *low*)
5	**if**(P is *high*) **and** (R is *high*)**and** (S is *medium*) **then** (CH is *high*)
6	**if**(P is *medium*) **and** (R is *high*)**and** (S is *medium*) **then** (CH is *high*)
7	**if**(P is *high*) **and** (R is *medium*)**and** (S is *high*) **then** (CH is *medium*)
8	**if**(P is *high*) **and** (S is *high*) **then** (CH is *medium*)
9	**if**(P is *low*) **and** (S is *high*) **then** (CH is *low*)
10	**if**(R is *high*) **and** (S is *high*) **then** (CH is *high*)

$$COG = \frac{\int_a^b \mu_A(x)xdx}{\int_a^b \mu_A(x)dx} \tag{4.4}$$

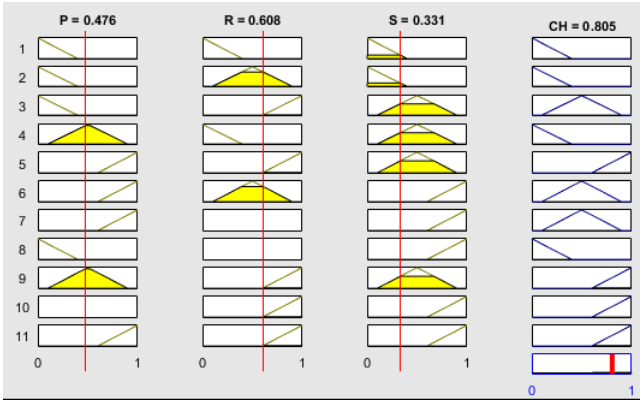


Figure 4.15: The output of the Defuzzifier

4.3.3.5.3 Data Block Structure: Health Data Block is presented in Table 6.6. The Block is divided into two parts- header and contents. The header holds metadata for contents.

4.3.3.5.4 Block Validation: The leaders selected by Algorithm 6 verify a new Block before sending the Block to the FAR processing layer. The verification process is depicted in Algorithm 10 where leader Agents check the hash value of the immediate previous Block, the integrity and signature of all the transactions packed in Merkle Tree of the Block. The leader first broadcasts the Block throughout the NEAR processing layer. If specific numbers of leader certify the Block as a valid Block, Agents in the NEAR processing layer sends the Block to the Agents in the FAR

Table 4.4: The Format of Data Block

Block Header of Blockchain	
Field	Description
Version	Block Version Number
Block Types	It indicates diverse types of Blocks(health record, financial record, diagnosis record)
Previous Block Hash	This field contains hash code of the previous confirmed Block
Timestamp	This filed records the Block creation time
Merkle Tree Root	Transactions are packed into a Merkle tree and this field stores the root of the tree
Vote	Miner verifies the Block and votes

processing layer for permanent storage. The Agents in the NEAR processing layer store metadata about a confirmed Block.

4.3.4 Security Protocol for the Decentralized Patient Agent

The same Agents replicated at the NEAR processing layer, and FAR processing layer require a secure communication channel to transfer health data between them. In this IoT eHealth, we used some standard security protocols to enforce security for the Patient Agents. The security protocol for eHealth architecture is described below.

4.3.4.1 Digital Signature

The Bitcoin or Ethereum Blockchain uses the PKI(Public Key Infrastructure) as an address for a user or Miner. Blockchain users generate a digital signature using PKI. The PKI can preserve a node's pseudonymous properties, but a user's transactions can be correlated to each other while the Miners verify those. In eHealth, a patient's privacy has the risk of being compromised if an attack can connect his sensitive health data to an Agent of the patient.

Ring Signature[457] can be an alternative to traditional PKI digital signature to preserve a pateint's privacy. A transaction's owner can form a digital ring signature by merging a group of other users' signature. As a result, an attacker cannot identifiy the owner of the data transaction because multiple entities participate in forming a ring signature. In our eHealth, the Patient Agent executing in the sensing layer is the signer and other neighbouring Agents at the NEAR processing layer are the ring members.

The Ring Signature's format depicted in Figure 6.10(c) is represented as $(m, P_1, P_2, ..., P_r; v, x_1, x2, ..., x_r)$ where $P_{1 \leq i \leq r}$ is the public key of the ring members, and and $x_{1 \leq i \leq r}$ is a random number selected by the signer(the Patient Agent in smartphone/Fog) for $P_{1 \leq i \leq r}$. m is signified as the original message for which the digital signature needs to be generated and v is the generated message(digital signature) that needs to be verified. A typical Ring Signature is generated as follows;

- **Message Digest:** The signer(data owner) generates k = H(m) and a random value(u). The signer performs a symmetric encryption on u with key(k) to produce v = Enc(k, u).

Algorithm 7: Block Validation Procedure

```
Data: Block(B)
Result: Confirmed Block
initialize statusSig = false, statusContent = false, statusPrevBlockHash = false, countVote = 0
super leader organizes a certain number of transactions into a Block
super leader sends the Block to leaders for the approval
for each leader k = 0 to l do
    while newBlockRequest = true do
        statusSig ← verifySignature(B);
        statusContent ← verifyIntegrity(B);
        statusPrevBlockHash ← verfiyPrevHash(B);
    end
    if statusSig == true ∧ statusContent == true ∧ statusPrevBlockHash == true then
        countVote + +
    end
end
for each general Edge Agent i = 1 to m do
    if voteCount ≥ thresholdCount then
        Block is transferred to the Cloud Blockchain
    else
        Block is rejected
    end
end
```

- **Signature Merge:** $e = x_i^{P_i} (\mathrm{mod}\, N_i)$ is calculated for each ring member except the signer. Here, x_i is a random number picked by the signer for the i^{th} member and P_i represents the public key of the member nodes. The signer also produces $v = v \oplus e$ for each member. The signer calculates $x_s = (v \oplus u)_d (\mathrm{mod}\, N_s)$ where d is the secret key of the signer.
- **Complete Signature:** Finally, the signer forms the signature as $(m, P_1, P_2, ..., P_r; v, x_1, x2, ..., x_r)$

The health data transactions to be processed in the Blockchain contains a ring signature. The Blockchain nodes check the data integrity using the ring signature. But, Blockchain Miners cannot trace the transaction's creator and distinguish the creator from other ring members.

4.3.4.2 Authentication between replicated Patient Agent

The Patient Agents at the three levels need to perform authentication using a session key every time they exchange medical data. The replicated Patient Agents(homogenous Agents) in the smartphone, Fog and Cloud dynamically come up with the same session key using DES Round Key generation algorithm[458] depicted in Figure 6.10(b) to avoid a man in the middle attack that occurs during the exchange of session key. A primary key for generating session key is inserted into the replicated Patient Agent during installation. Later, the generation of the session key is achieved by using the same algorithm. Suppose, K_s^i is the session key generated by the three replicated Patient Agent for the i^{th} session. The homogeneous Patient Agents need to exchange $HMAC(K_s^i, Time||Id_i)||Time$ depicted in Figure 6.10(a) to authenticate each other. Key Exchange between heterogeneous replicated Patient Agents occurs using Diffie-Hellman key exchange method[459].

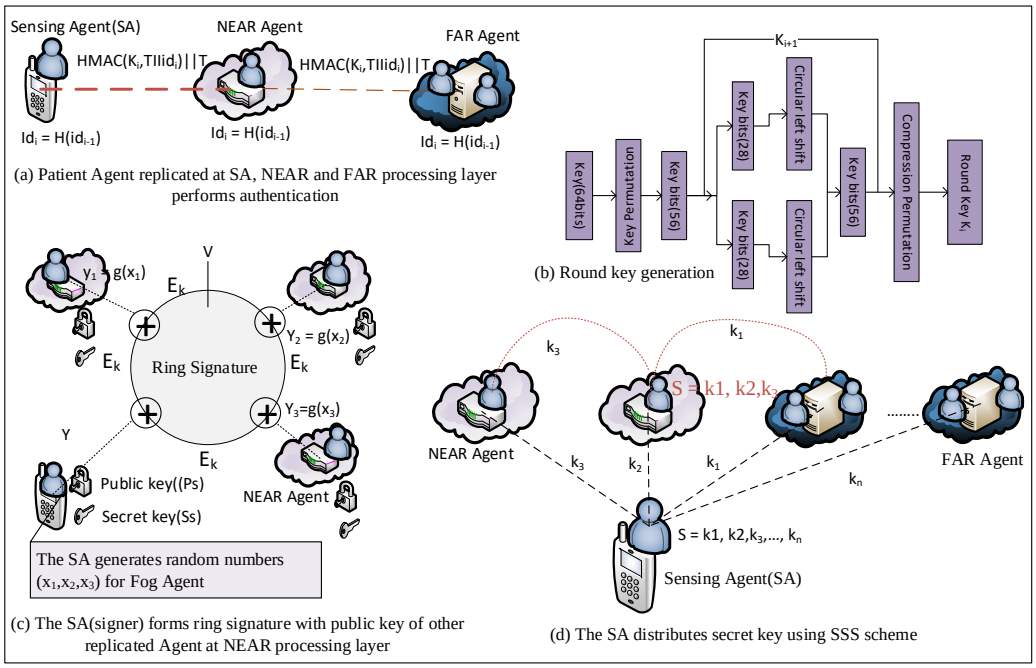


Figure 4.16: The security method for the framework

4.3.5 Data Encryption Key Management

Symmetric key encryption technique such as AES or DES[458] is appropriate for memory and power-constrained devices like smartphones[54]. The Patient Agent replicated at three levels: smartphone, Fog and Cloud each store asymmetric data encryption key or exchange the key. The data encryption key might be compromised by a malicious hacker or by a rogue Fog or Cloud administrator. The replicated Patient Agent needs to secure their keys. There needs a key management protocol that does not allow any Patient Agent to obtain the key without the approval of other replicated Patient Agents. The SSS(Shamir's Secret Sharing) scheme depicted in Figure 6.10(d) is utilized to distribute a secret key among the replicated Patient Agents. The secret key(S) to decrypt data is divided into pieces of data $S_1, ..., S_n$ and each piece is shared with n replicated Patient Agent accordingly.

1. A Patient Agent requires knowledge of k or more S_i from other replicated Patient Agent to compute the complete secret key S. For instance, if the Patient Agent is replicated in five different devices, k might be two or three.

2. A Patient Agent cannot reconstruct the secret key S with fewer than k pieces. S remains completely undetermined with knowledge of k – 1 or fewer S_i pieces

This key sharing scheme is called (k, n) threshold. Every time, a Patient Agent requires to decrypt data, it asks k – 1 pieces of S_i from other replicated Patient Agent to make the complete secret key(S). This scheme prevents the attack from gaining unauthorized access to keys, even if the device is compromised.

4.4 Performance Analysis

In this section, we discussed and analyzed the performances of the key algorithms of the proposed architecture. The simulation is coded using Java Programming following iFogSim[128]. Table 6.7 presents the simulation parameter.

Table 4.5: The Parameters for the simulation

Network Area	1000×1000m^2
Device Radio Range	300m
Fog device CPU capacity	9900MIPS - 83000MIPS(Million instruction per second)
Smartphone CPU capacity	14000 MIPS
Fog device RAM capacity	8 - 16
Fog device Bandwidth	600 - 300Mbps
Smartphone Bandwidth	100-50Mbps
Fog device Power Consumption Rate(per Hour)	140-95W
Fog device Transmission Power Consumption Rate(per Hour)	10W
Smartphone Power Consumption Rate(per Hour)	25-20W
Smartphone Transmission Power Consumption Rate(per Hour)	2
Transaction Size	1024 bytes
Block Size	10× 1024 bytes
Size of the tasks to be migrated	10-5KB/MB
Instruction required to validate Block	10Million
Instruction in a Task	100-50Million

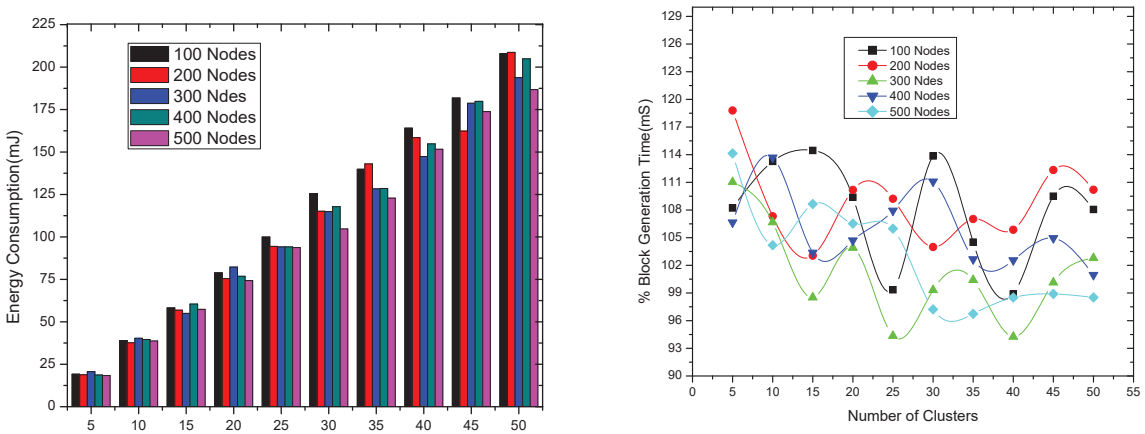
4.4.1 The Consensus Mechanism:

The nodes in the simulation are located in $1000 \times 1000m^2$ area. Performance of the consensus mechanism is estimated considering a variable number of nodes 100, 200, 300, 400 and 500 and a variable number of clusters such as 5, 10, 15, 20, 25, 30, 35, 40, 45 and 50 respectively for each group of nodes. The member nodes within a cluster can directly send or receive data to/from a cluster head. But, nodes within the inter-cluster communicate using the shortest path routing such as Dijkstra's algorithm. The performances of the modified PoS consensus mechanism and the standard PoS are investigated for the following parameters.

- **Energy Consumption:** Energy consumption refers to the energy required for transmitting, receiving transactions, validating a certain number of Blocks on the simulated network.
- **Block Generation Time:** This refers to the time required for transmitting, making Blocks and validating a certain number of Blocks on the simulated network.

The modified PoS consensus mechanism is executed ten times in the simulated network, and the performance graphs are depicted with average values generated from 10 execution runs. The standard PoS runs on a horizontal network, and the modified PoS is designed to run on a hierarchical network. For both kinds of consensus mechanisms, nodes that lock digital coin to the system participate in mining. Figure 4.17 depicts the consumption of energy and execution time to generate 100 number of Blocks providing that a variable number of nodes and clusters are considered.

The graph depicted in Figure4.17(a) shows that energy consumption proportionately increases with an increasing number of cluster heads in the network because the cluster head plays the role of validating Blocks. Further, cluster formation using K-means and cluster head selection algorithm consumes power. In contrast, the graph demonstrates an almost similar amount of power consumption regardless of the number of nodes for a particular cluster. This is significant advantage of running consensus mechanism in the hierarchical network.



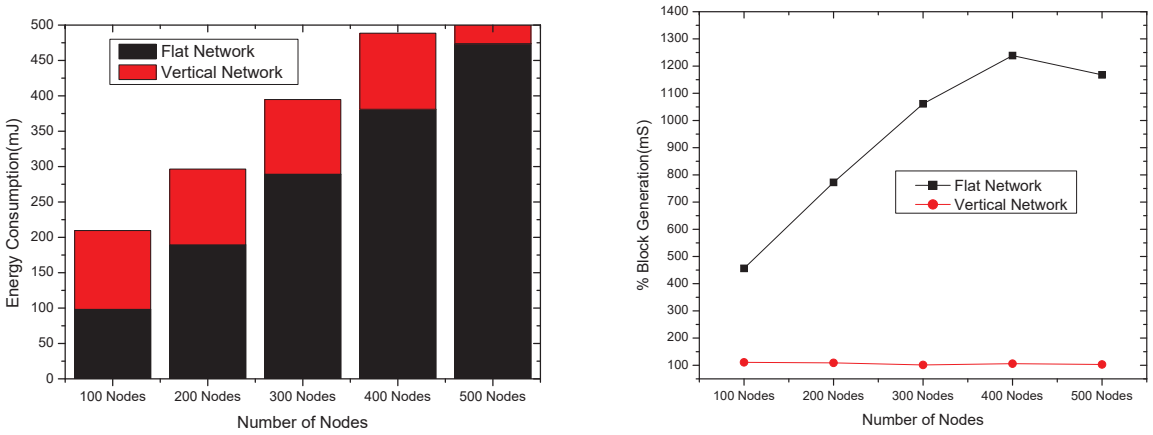
(a) Clusters vs Energy Consumptions

(b) Clusters vs Throughput

Figure 4.17: Performance of modified PoS mechanism in terms of energy consumption and Block generation time

Figure 4.17(b) shows the Block(100%) generation time for different number of clusters and nodes. The graph depicted in Figure 4.17(b) shows that Block generation time with a higher number of clusters does not follow a consistently lower or higher trend. Cluster heads gather transactions and make Blocks; so higher number of Blocks are generated per second with a higher number of clusters. On the other hand, higher Block generation time was also found for some higher numbers of the cluster due to delay in verifying Blocks. This indicates that a standard number of cluster needs to be determined to have better outcomes. The ideal number of clusters vary depending on the number of nodes. For instance, the ideal number of cluster for 300 nodes is 25 but it is 35 for 500 nodes.

The performance of the modified PoS is compared with the standard PoS in terms of energy consumption and Block generation time. In Figure 4.18(a), the modified PoS shows a significant reduction of energy consumption compared with the standard PoS. In the modified PoS, few selected healthy Miners validate a Block, but the standard PoS requires more than 50% node participates in the Block validation process, which results in higher energy consumption. Energy consumption of modified remains almost constant for a particular number of clusters with a higher number of nodes, whereas energy consumption of PoS keeps increasing when the number of nodes in the network is increased.



(a) Nodes vs Energy consumption

(b) Nodes vs Block generation time

Figure 4.18: The comparison of performance between modified PoS and standard PoS in terms of energy consumption and throughput

The Block generation time of modified PoS and standard PoS is demonstrated in Figure 4.18(b). The graph depicted in Figure 4.18(b) shows that the Block generation time standard PoS is higher than modified PoS. In standard PoS, different nodes transmit their transactions to one leader node and confirmed Block is needed to broadcast throughout the network for validation. Consequently, the process consumes higher energy, as depicted in Figure 4.18(a) and requires a longer time for broadcasting the Block throughout the network. Further, the modified PoS selects some healthy Miners based on reputation, performance and stake, but standard PoS nominates a Miner based only on investment or stake. Therefore, Block generation time is lower in the cluster-based network with modified PoS.

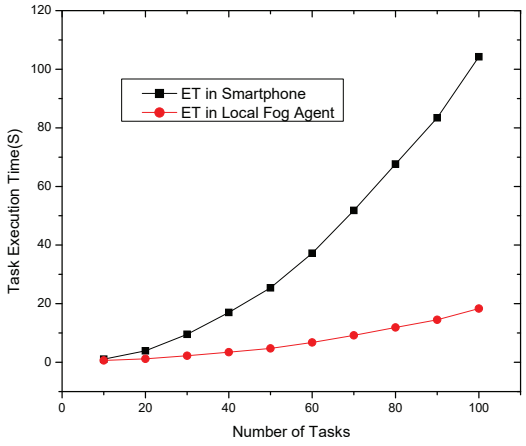
4.4.2 Task Migration Algorithm:

The Blockchain leveraged task migration approach was also executed in the above mentioned simulated network. The Hungarian assignment algorithm was implemented using Java Programming. We analysed the performance of the task migration methods with respect to variable numbers of tasks such as 10, 20,30,...,100. The task migration algorithm was run by smartphone Agent and Fog Agent. The smartphone transmits tasks to the local Fog Agent. The local Fog Agent utilizes Blockchain to transfer the tasks to other neighbouring or remote Fog Agents. Every Agent applies FCFS(First come First Service) as CPU scheduling to process their jobs. The performance of the task migration method is discussed in terms of the following two metrics.

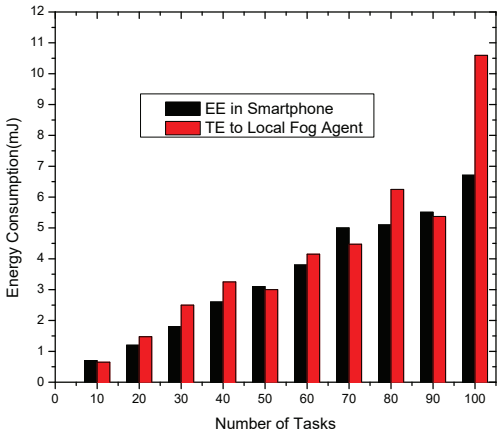
- **Energy Consumption:** This refers to the energy required for a local Agent to locally execute a task or transmit the task to a foreign/remote Fog Agent.
- **Execution Time:** This refers to the time required for a local Agent to execute a task locally or receive a response for the task from a remote Agent.

EE, ET, TE and TT in figure 4.19 and 4.20 are acronyms of Execution Energy, Execution Time, Transmission Energy and Transmission Time, respectively.

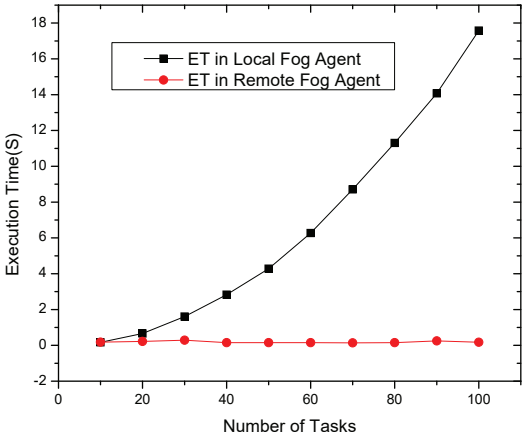
The graph for a variable number of tasks vs execution time and a variable number of tasks vs energy consumption are depicted in figure 4.19 for the smartphone Agent and local Fog Agent



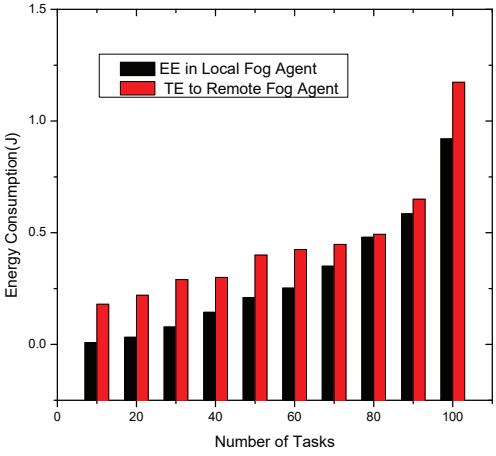
(a) Tasks Number vs Execution Time (smartphone Agent)



(b) Tasks Number vs Execution Time (smartphone Agent)



(c) Tasks Number vs Execution Time (Fog Agent)

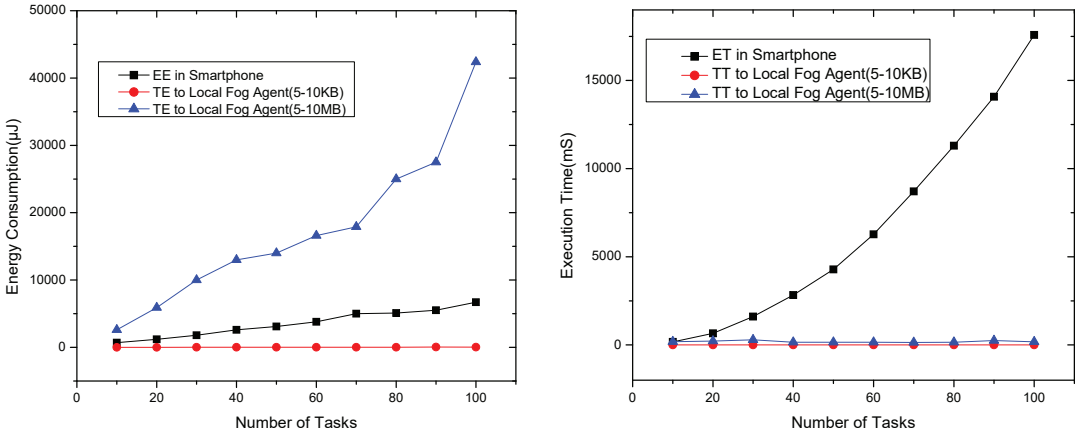


(d) Tasks Number vs Execution Time (Fog Agent)

Figure 4.19: The response time and energy consumption for local execution and transmission

when they execute a set of heavyweight tasks(5-10 MB). The processing capabilities of a local Fog Agent are higher than that of a smartphone Agent. As a result, the graph in figure 4.19(a) shows that the completion time of every set of tasks required in the local Fog Agent is less than that of completion time in the smartphone Agent. The smartphone demonstrated longer queue delay than local Fog Agent in the simulation environment. On the other hand, the graph in figure 4.19(b) shows that smartphone consumes more energy or hardly a bit less energy(for the number of tasks 50 and 70) to transmit heavyweight tasks to the local Fog Agent than to execute those locally. The graph shows that the smartphone cannot save energy if the tasks are offloaded to the local Fog Agent.

Figure 4.19(c) demonstrated that if the local Fog Agent outsourced tasks to neighbouring or remote Agents, the overall response time for those tasks is reduced. The foreign Fog Agents parallel execute the assigned tasks. On the other hand, figure 4.19(d) depicts that the local Fog Agent had to spend higher energy to transmit tasks to foreign Agents than to execute those tasks locally. Consequently, tasks should be divided as delay sensitive or energy sensitive.



(a) Tasks vs Energy Consumpiton(smartphone)

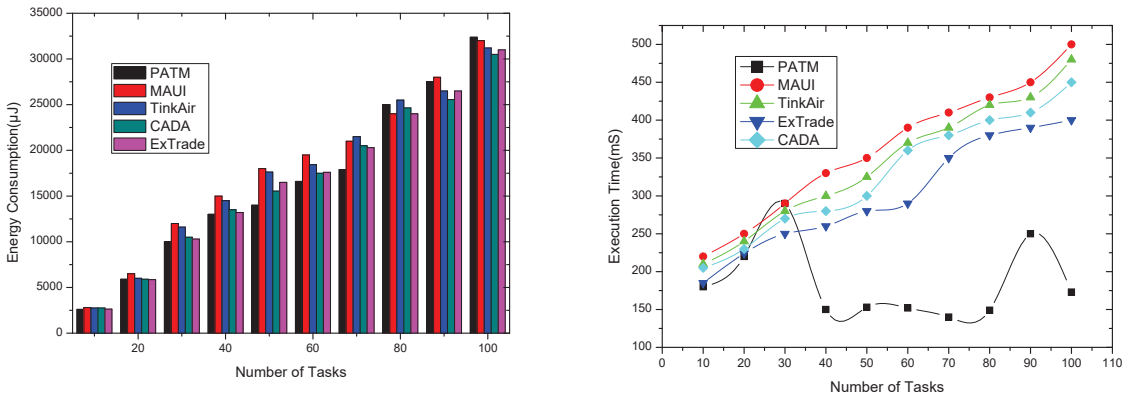
(b) Tasks vs Execution time(smartphone)

Figure 4.20: The comparison of performance for lightweight and heavyweight tasks

The energy consumption and time for data transmission to remote devices depend on the task's size. The smartphone transmits two kinds of tasks: lightweight(5-10KB) and heavyweight tasks(5-10MB) to the embedded Fog Agents. The effect of task's data size on execution time and energy consumption is shown in figure 4.20(a) and 4.20(b). Migrating lightweight tasks needs less energy consumption than that of heavyweight tasks. The smartphone benefits lower transmission energy consumption and time if the transmitted task's size is small.

The energy consumption of five offloading approaches is depicted in Figure 4.21(a). This energy consumption includes the energy required for a task's transmission and execution. The proposed Patient Agent-based task migration(PATM)improves energy consumption over other methods when the number of tasks is not many. The PATM consumed high energy for the larger number of tasks because the Hungarian assignment algorithm costs much in terms of energy and time for a large number of tasks. Overall, the PATM saves 1.81% and 8.45% energy in comparison to ExTrade and MAUI approaches, respectively.

The comparison of the execution time among five offloading approaches is depicted in Figure 4.21(b). The PATM improves the execution time over other approaches because the Hungarian method chooses some remote Fog devices to optimize the execution of all the tasks. Other migration approaches serially assign a task to a remote Fog device. Other approaches show higher



(a) Tasks Number vs Energy consumption

(b) Tasks Number vs Execution time

Figure 4.21: The comparison of performance among few offloading approaches

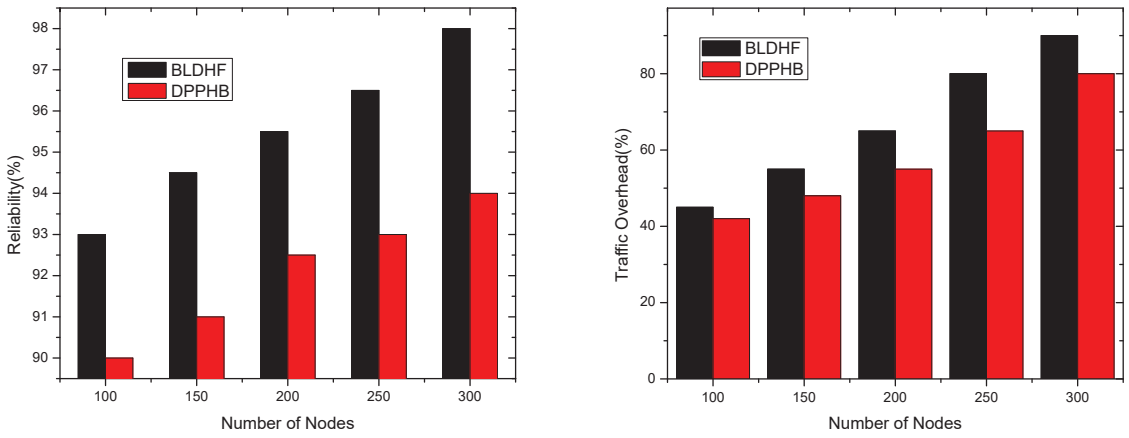
execution time as the number of tasks is increasing, whereas the PATM shows almost constant execution time for the increasing number of tasks. The PATM not only decides to offload but also optimally assigns tasks to remote Fog Agents. Overall, the proposed tasks assignment improves execution time 38.28% over the ExTrade approach that shows the lowest execution time among the existing methods.

4.5 Security Analysis

The code for the ring signature and secret key sharing module are downloaded from [457] and [460], respectively. We compared our eHealth architecture(BLDHF) with an existing eHealth architecture(DPPHB) with respect to reliability and communication overhead. These two performance metrics are related to the security protocol.

Communication Overhead Issue: Communication overhead: Communication overhead refers to data transfer overhead. To transfer a payload of data over a communication channel requires sending more than just the payload itself. Extra information includes various control packets and signalling data. Communication overload is measured as the percentage of non-application bytes divided by the total amount of bytes in the packet/message. In our simulation setting, we assessed the communication overhead required to transfer data between Patient Centric Agent at three levels. Assessing communication overhead for the entire PCA assisted decentralized Blockchain network is subject to our future research.

The graph depicted in figure 4.22(a) shows that our eHealth achieved higher reliability than DPPHB(A Decentralized Privacy-Preserving Healthcare Blockchain for IoT) because of our decentralized key management and multiple instances of the Patient Agent at three layers. On the other hand, the graph depicted in figure 4.22(b) displayed that the security mechanism in our eHealth caused higher communication overhead than DPPHB. In our system, an Agent requires to collect a certain number of segments of the data encryption from other neighbouring Agents to form the complete secret key. This method causes communication overhead while authenticating and exchanging secret keys.



(a) Nodes vs Reliability

(b) Nodes vs Overhead

Figure 4.22: The comparison of performance between two eHealth architectures

We also used Scyther[129], a formal methods tool to verify the authentication process. Scyther measured the strength of the authentication protocol in our architecture against security attack. Figure 4.23 shows the outcome of our authentication protocol in Scyther. The automated process of these tools provides checking for authentication, secrecy and message integrity. Scyther analyses the performance of security protocol regarding the following parameters.

1. Alive: Scyther can test the aliveness of the communication parties so that they can perform events successfully and be available at any time. This indicates the analysis of a DoS attack.

2. Secret: Secret means that the role is secret and there are no attacks within bound, data will require a parameter term to verify the claim. The user needs to pre-set the parameters before testing the claim. This proves that the proposed protocol is protected and ensures the confidentiality of data is provided.

3. Nisynch: It is a No-injective Synchronization. This term ensures the successful synchronization, no reply attack, and mutual authentication. This term is used to check if the security protocol safeguards against the replay attack.

4. Niagree: The integrity of data can be verified by using the non-injective agreement on message. The term ensures that the original data from the legitimate source is not modified over the communication channel.

Figure 4.23 shows **OK** for the above claims, which indicates that the applied authentication protocol can withstand different kinds of security attacks.

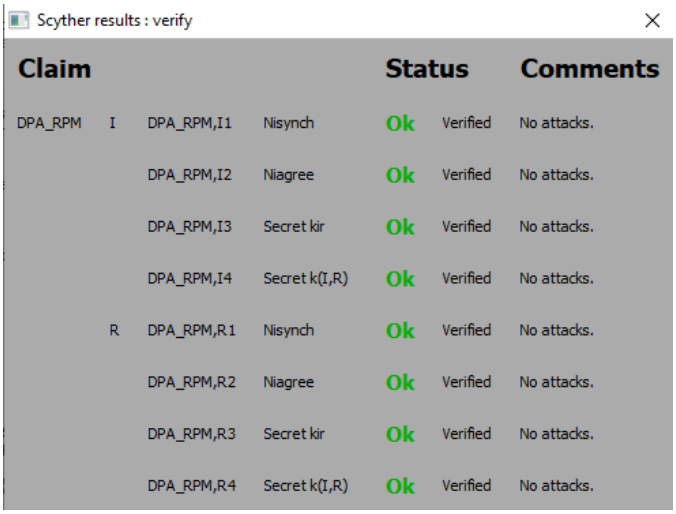


Figure 4.23: The attack result from Scyther tools

The architecture needs to be discussed in terms of basic security requirements: Confidentiality, Integrity and Availability.

1. Confidentiality: Like Cloud, heterogeneous Fog devices with diverse security methods or no security are deployed by different stakeholders. Sensing and processing of health record by Fog devices is susceptible to malicious attack. In our architecture, the same Patient Agent for a patient replicated in the Smartphone, Fog device and Cloud can safeguard health record from malicious attack. The sensitive medical data is analyzed in the homogeneous replicated Patient Agent to preserve patient's privacy or confidentiality. Further, patient's record stored in Blockchain decentralized ledger distributed among multiple public servers encounters the potential privacy leakage because patient has to provide the server with his or her private/public key to decrypt the ciphertext while retrieving health data. This retrieval results in a potential privacy leakage[83]. The replicated Patient Agent can protect the patient's privacy while retrieving health data from the public domain if an individual Agent dedicated to a Patient for creating and managing keys is installed in the public server. Further, a patient's identity is completely anonymous in the Blockchain because of using a ring signature.

2. Integrity: The Fog Agent stores Block's header in the fashion of linked list, which ensures health data integrity.

3. Availability: There is multiple Patient Agent at different levels to process health data and multiple Cloud server to store health data that facilitates the access of health data from multiple points. Therefore, the architecture is providing the users with high availability.

Furthermore, the security strength of the proposed decentralized eHealth architecture is discussed in terms of some attacks, including DoS, mining, storage, and dropping attack[91]. The attack description and mitigation are illustrated here.

1. Dropping Attack: A Dropping Attack occurs when a cluster head drops the transactions. But this is unlikely to happen because the cluster head will lose its reputation and share once it is identified as malicious. If the cluster head is down or malicious and cluster members do not receive transactions for verification, the consensus mechanism should select another node as the cluster head. The cluster member temporarily stores transactions until the Block containing the transaction is confirmed in the Blockchain. Therefore, lost transactions can be retrieved from the cluster members.

2. Storage Attack: A group of malicious nodes can store and corrupt the Blockchain ledger and make heath record inaccessible to intended parties. The Blockchain ledger is stored in the Cloud server. Patient Agents for different users use different Cloud servers. Many Cloud server contains the exact copy of the complete Blockchain ledger. The Fog devices also store a chain comprising Block's header without data, required to prove the integrity of the ledger. Data can be retrieved even if some Cloud servers corrupts the ledger because the headers stored in Fog can be used to reinstate corrupt Blocks. This makes a Storage Attack unlikely to be successful.

3. Mining Attack: A 51% attack is called a mining attack where more than 50% nodes can control the network. We divide the entire network into clusters, and the cluster head is responsible for collecting transactions from that cluster members. The cluster head is changed depending on the performance, reputation and locked share after a certain period. A super leader is randomly chosen from the cluster heads. So, nodes from a particular region cannot collude for a mining attack.

4. Denial of Service attack: Denial of Service attack means to shut down the usual activities of a machine through flooding unwanted traffic, causing the legitimate user unable to access the machine. In our eHealth architecture, the Patient Agent is replicated at three different levels. The Patient Agent executing in the smartphone resumes the services through replicating a Patient Agent at another device at Fog or Cloud level when the Patient Agent is under the DoS attack. Further, a node needs to lock its coin to participate in mining. A user needs to pay a transaction fee to include the transactions in the Blockchain. The locked coin and transaction fee safeguard the system from a DoS attack by the registered nodes.

5. Selfish Mining: Selfish Miners attempts to increase their share by not broadcasting mined blocks throughout the network for some period and then releases several Blocks at a time making other Miners lose their Blocks. With our PoS, a super leader can make a certain number of Blocks. In every round, the new super leader is randomly selected to organize transactions into a Block. This approach can reduce the possibilities of such an attack.

Privacy Issue of Consensus Protocol: To ensure end to end security, the PCA encrypts all packets using symmetric key exchanged using Diffie–Hellman (DH) Algorithm. Further, Ring signature has been employed to ensure privacy of the Patient Centric Agent that participate in the consensus process. We will incorporate TLS for ensuring end to end security for the proposed consensus mechanism.

The Table 4.6 presents‘ the comparative analysis of our architecture with other existing Blockchain based frameworks.

Table 4.6: The comparative analysis of the our eHealth system with existing systems

Comparison Criteria	Proposed eHealth system	Existing system - 1[417]	Existing system-2[89]
Fault tolerance	High, multiple instances of a PA at three layers manage health data	High for the Blockchain Multi-access Edge Network but low for the sensor network	Medium, single agent controls and manage Blockchain
Confidentiality, Integrity and Availability(CIA)	High CIA, homogeneous PA processes sensitive medical data using ring signature, Edge nodes maintain Blockchain for metadata, multiple PAs ensures service availability	low confidentiality due to third parties' involvement in processing health data, integrity High because of using Blockchain, availability limited due to centralized broker	confidentiality is medium, integrity is high, availability is low due to centralized Blockchain controller
Cyber Attacks	Withstand Ransomware, and DoS attacks	Local processing unit and a universal broker are vulnerable to Ransomware and DoS attack	centralized PA is vulnerable to many cyberattacks
Data Immutability	Yes	Yes	Yes
Secure and energy efficient migration	A privacy aware Blockchain leveraged task migration method	No such approach was designed	No such approach was designed
Interoperability	Yes	Yes	Yes
Scalability	Medium, many resources are required	High	High
Service Reliability	High	Medium	Medium
Consensus Mechanism	Lightweight consensus mechanism was proposed	Exiting Proof of Work(PoW) consensus mechanism was used, high computational cost	Modified PoW, medium level of computational cost
Communication Overhead	High traffic due to exchange security keys	Low because security management module was not included	Medium, limited exchanges of security key

4.6 Conclusions

In this paper, we designed an eHealth system that deployed multiple instances of a software Patient Agent at three layers: Sensing, NEAR processing and FAR processing layer. This makes the eHealth system more reliable and fault-tolerant. We also described how the Patient Agent could be adopted on a 5G architecture. The dedicated Patient Agent software can manage the resources of 5G network slices to embrace the Blockchain technologies in processing health data. The eHealth system includes a modified Blockchain PoS consensus mechanism and a privacy-aware task offloading algorithm. In this eHealth architecture, homogeneous Patient Agents(instances of the same Patient Agent) use digital ring signature and SSS(Shamir's Secret Sharing) to ensure secure communication channel between them. A performance analysis demonstrated that the proposed eHealth system could perform the processing of health data in near real-time using Blockchain technologies. The adoption of Blockchain technologies in healthcare is challenging with a massive amount of health data continuously streamed from wearable sensors. Not all medical data generated from continuous patient monitoring needs to be stored with the same security and privacy level. Health data can be disseminated among multiple health repositories(EHR, EMR, PHR and Blockchain EHR) in accordance with patient's privacy preferences. Our future work is to develop a dynamic storage selection algorithm soliciting patient's preferences regarding his or her privacy and security.

Health data is not uniform, and varies at each point in the degree of significance, privacy, and security required. Continuous tracking of patients produces large amount of data from wearable sensors. Blockchain technology is questioned as an electronic health record system, with limited storage space and low throughput. Storing all kinds of health data in the Blockchain pressures the Blockchain participants' storage capacity and the network suffers from poor efficiency and throughput. To address this issue, we explored a few other health repositories in the next chapter, and developed a model to suggest health data taking into account data requirements such as privacy, security and quality of service.

Chapter 5

The PCA Managed Rapid Storage Allocation of IoT Health Data with a Machine Learning Model

Most Blockchain (BC) researchers have recommended off-chain repositories to store IoT data and on-chain to save the hash pointer of the data. This strategy can significantly reduce the burden of storage on the Blockchain nodes. However, this policy goes against the tamper-proof nature of the Blockchain and undermines the strength of this technology. Blockchain technology can eliminate the need for third parties to verify and process transactions because every Blockchain miner owns a local ledger. This obviates the need to request a centralized server for transaction's verification. Storage of data in the conventional database and managing their hash pointer on the Blockchain cannot guarantee document alterations. Blockchain holding a hash pointer to the data can only detect if data has been changed or not.

To resolve this problem, we proposed to classify health data, particularly continuous health data as normal or abnormal or relevant/irrelevant in the previous works. Abnormal and relevant health data are uploaded on the Blockchain ledger, and normal data might be stored in less secure repositories such as Cloud servers. Transactions from abnormal or relevant health data are usually less frequent than normal/irrelevant data. This approach partially addresses the health data storage problem on the Blockchain. However, health data is exponentially expanding and not uniform; instead, they vary in the levels of security, privacy they require, the quantum of storage, the genre of health data, and quality of services.

Recently, a wide range of digital health record repositories has emerged. These include Electronic Health record managed by the government, Electronic Medical Record (EMR) managed by healthcare providers, Personal Health Record (PHR) managed directly by the patient and new Blockchain-based systems mainly managed by technologies. Health record repositories differ from one another on the level of security, privacy, and quality of services (QoS) they provide. Health data stored in these repositories also varies from patient to patient in sensitivity, and significance depending on medical, personal preference, and other factors. Decisions regarding which digital record repository is most appropriate for the storage of each data item at every point in time are complex and nuanced. The challenges are exacerbated with health data continuously streamed from wearable sensors.

To address this issue, we expanded our previous idea. We introduced a machine learning solution to disseminate health data among multiple health repositories by mapping the data storage

requirements with the features of health repositories. For instance, the scheme chooses BC ledger for the sensitive and lightweight health data such as prescription, diseases summaries, medication, and diagnosis report and billing transactions that require a distributed platform for processing without the aid of third parties. Similarly, the scheme recommends proprietary Cloud repository for storing a non-sensitive and high volume of data such as non-identifying ECG readings.

Diverse health repositories are associated with different levels of privacy, security and QoS. Many researchers [461–463] have designed methods for choosing appropriate Cloud Service Providers (CSPs) to store consumer data, taking into account the performance and cost parameters of the CSPs. However, there is no state-of-the-art mechanism to recommend the storage repositories for diverse kinds of medical data particularly streaming medical data, taking into account patient's preferences for security , privacy, cost and other data related features including QoS, data sensitivity, quantity of data and data type.

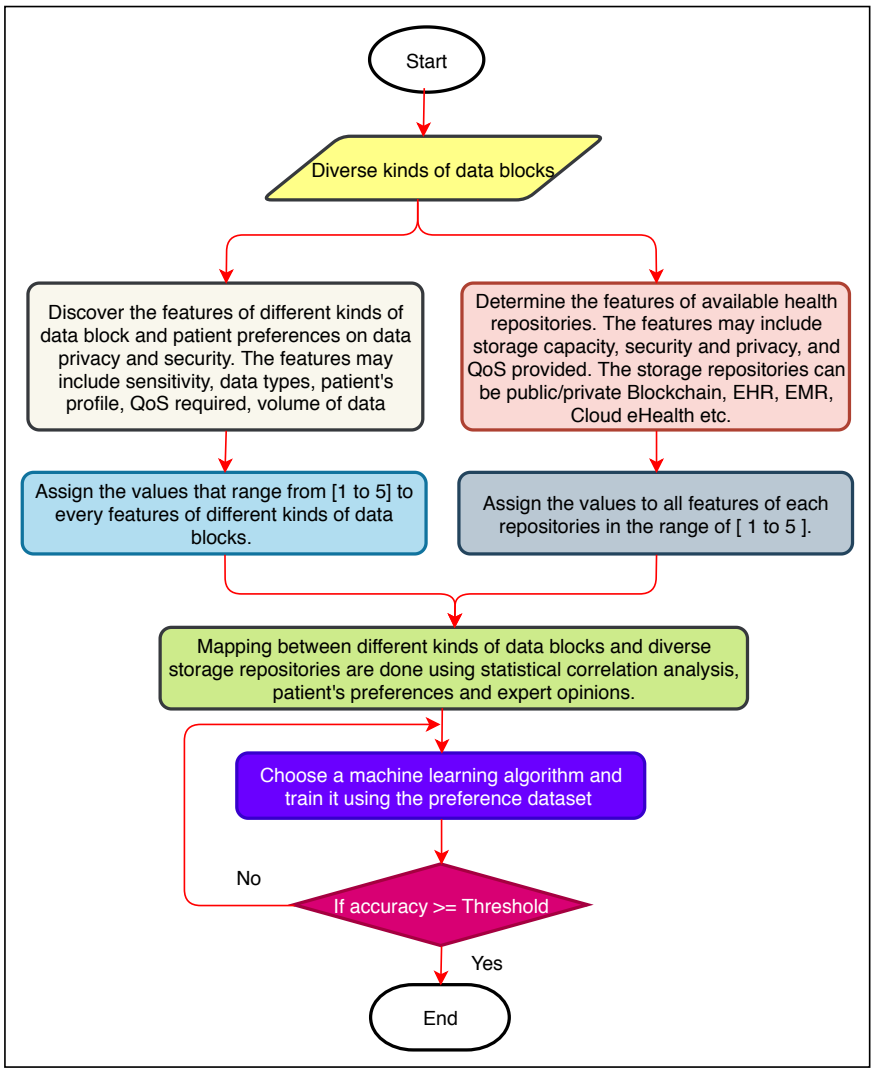


Figure 5.1: The flow diagram of the proposed recommendation model

To cope up with the above mentioned issue, in this chapter, we described a recommendation model for health data storage that can accommodate patient preferences and make storage decisions rapidly, in real-time, even with streamed data. The model maps health data to diverse repositories

so that the storage medium can satisfy data block requirements. The mapping between health data features and characteristics of each repository is done using a machine learning-based classifier mediated through correlation analysis, user preferences and clinical heuristic rules. The flow diagram of the IoT health data storage model is depicted in Figure 5.1. According to Figure 5.2, the first step of the model is to determine the storage requirements of data blocks and the metrics to assess different health record systems. In the next phase, both data storage requirements and parameters for evaluating repositories are normalized in the range of [1 to 5]. Thirdly, the association between data storage requirements and repositories' metrics is analyzed, which leads to forming a training dataset for the machine learning algorithms using correlation analysis, user preferences and heuristic rules. Evaluation results generated from Weka tools and a deep learning model using Python with Keras framework demonstrate the model's feasibility.

The model can be an important tool for improving storage and access arrangements with the exponential growth in the amount of health data that needs to be stored and accessed globally. The approach can enable patients to control the storage and access of their health data, while also ensuring that data storage are manageable from a size perspective.The ML model accommodates the storage solution most 'fit for purpose' for various data properties.

The contents below of this chapter were published in the **Health Informatics Journal**, SAGE in September 2020. The current impact factor of the journal is 2.923
M. A. Uddin, A. Stranieri, I. Gondal and V. Balasubramanian, " Rapid Health Data Repository Allocation using Predictive Machine Learning," Health Informatics Journal, SAGE, 2020. https://doi.org/10.1177/1460458220957486.

Abstract

Health-related data is stored in a number of repositories that are managed and controlled by different entities. For instance, Electronic Health Records are usually administered by governments. Electronic Medical Records are typically controlled by health care providers, whereas Personal Health Records are managed directly by patients. Recently, Blockchain-based health record systems largely regulated by technology have emerged as another type of repository. Repositories for storing health data differ from one another based on cost, level of security and quality of performance. Not only has the type of repositories increased in recent years, but the quantum of health data to be stored has increased. For instance, the advent of wearable sensors that capture physiological signs has resulted in an exponential growth in digital health data. The increase in the types of repository and amount of data has driven a need for intelligent processes to select appropriate repositories as data is collected. However, the storage allocation decision is complex and nuanced. The challenges are exacerbated when health data are continuously streamed, as is the case with wearable sensors. Although patients are not always solely responsible for determining which repository should be used, they typically have some input into this decision. Patients can be expected to have idiosyncratic preferences regarding storage decisions depending on their unique contexts. In this paper, we propose a predictive model for the storage of health data that can meet patient needs and make storage decisions rapidly, in real-time, even with data streaming from wearable sensors. The model is built with a machine learning classifier that learns the mapping between characteristics of health data and features of storage repositories from a training set generated synthetically from correlations evident from small samples of experts. Results from the evaluation demonstrate the viability of the machine learning technique used.

5.1 Introduction

The management of health data is no longer exclusively regulated by clinicians but increasingly requires a level of consent from patients[464]. Patients can decide who can access, analyze, and exchange their health information more than ever[465]. For instance, a patient has a great deal of control over Patient-Generated Health Data (PGHD) created, generated and collected by themselves, such as vital signs or fitness data[465]. Managing PGHD requires effort, cost and time for assimilating the data and as a consequence PGHD is rarely integrated with other repositories.

Patients who shared their self-tracking data with service providers expressed their dissatisfaction with the level of the provider's engagement with the data[466]. Despite this, PGHD can enhance medical care if the data can be incorporated with the current health data systems following data storage requirements. Broad categories of patient-generated health data[467] such as medication information, biometric tracking, behavioural tracking, environmental tracking, social interactions tracking, genetic information, mental health assessment, symptom tracking, reported outcomes, and legal documents have been identified in the literature[468]. However, few studies have examined the management of storage of various kinds of health data generated by patients.

Legislation has emerged in most jurisdictions regarding the storage of health data. Most legislation such as HIPPA[469] and AHPRA[470] are organization-centric legislation. Healthcare professionals or agencies typically own the health data that is produced and gathered under their oversight. However, some jurisdictions such as GDPR[471], [472] introduced in Europe, are consumer-centred regulations where the patient has complete control over health data and must consent to the collection of his or her health information, decide how long the health care professionals will hold the data, and where the collected data will be processed and stored [473]. Although the data protection regulations of the GDPR enable patients to have complete control over their health data, most users are unable to handle large quantities of data, understand the nature of the data collected, various methods of processing and track their personal data in compliance with the GDPR requirements[53, 471].

The appropriate management of health data is necessary to protect the patient's privacy and confidentiality while ensuring that data is available to relevant stakeholders. Recent reviews have identified the security of health data to be a major issue, particularly with the emergence of data from power, and memory limited medical sensors [474, 475] and huge medical data repositories[476]. Currently, the huge volume of health data is stored in repositories managed by different types of organizations. We discuss seven such health record agencies below:

1. **Governments Controlled EHR:** A government-managed electronic health record (EHR) is a record of a patient's health events throughout the lifespan. Diverse healthcare providers have access to subsets of the data where access is controlled by patients to different degrees. For instance, My Health Record[477] run by the Australian Government provides patients with mechanisms to control access to the data except in the context of criminal investigations or national security. EHRs are typically regulated by national laws that prescribe constraints such as the requirement that data be stored within national boundaries.

2. **Proprietary eHealth Cloud:** Global entities including Microsoft[478], Google[479] and Apple[480] have hosted health data repositories on publicly accessible Cloud storage medium. Though these global entities have struggled to maintain continuity of service, smaller-scale proprietary repositories are continuously emerging, offering public or private Cloud-based medical records storage. Patients are often provided with a high degree of control of their data by proprietary eHealth Cloud providers. However, aggregated data can be on-sold by

these providers to third parties, and Cloud administrators can always access confidential data.

3. **Technology managed Blockchain EHR**[54, 167, 270, 387] is a decentralized, tamper-proof ledger-based EHR in which a certain number of transactions are bundled into a Block to be reviewed by nodes called Miners prior to writing the Block in the current ledger.

   Some startup Blockchain-based EHR projects such as Patientory [481], and GEM[482] have recently been introduced. Access to data is completely controlled by patients with no exceptions for giving control for criminal investigations, system administrators or other entities. IBM estimated that 70% of healthcare leaders expect that Blockchain technology will enhance current clinical trial management, regulatory compliance, and promote a decentralised health record sharing system (HRS)[483]. Blockchain supports the processing and exchange of health data without the need for third parties trust. Traditional health record systems maintain a database that is operated and maintained by a single agency. In contrast, the Blockchain database is available to all individuals, but a user can access his or her information stored on the Blockchain. In Blockchain technology, miner nodes verify and validate transactions on a peer to peer network before committing those transactions in a ledger that is replicated amongst all participants of the system which guarantees the immutability and irreversibility of the recorded documents. Further, public cryptography applied in the Blockchain ensures data persistence, provenance, distributed data control, accountability and transparency. Blockchain leveraged health record systems can accelerate collaboration, sharing, integration of health data across various health agencies, healthcare professionals and patients[484, 485].

4. **Healthcare service providers Electronic Medical Records (EMR):** Most contact patients have with their provider leads to data being added to the provider's Electronic Medical Records(EMR). In most jurisdictions, providers own and control the storage and access to patient records though all providers must comply with regulatory requirements prescribed by acts such as the Health Records Act in Victoria[486]. Patients have varying levels of access to data stored in repositories managed by healthcare providers.

5. **Insurance organizations Health Database:** Health records are often stored in repositories controlled by insurance agencies or related organizations. Patients typically have minimal control or access to data managed by insurance agencies.These health databases are mainly managed for billing and administration, but can also be utilized by researchers, health authorities and other stakeholders to promote observational studies.

6. **Disease specific registries:** Registries for cancer-related records were first launched in North America and Europe between 1940 and 1950 respectively[487]. The cancer registry holds studies, screening, and test findings related to various cancers such as skin, breast, and cervix, as well as malignant tumours to provide information on the occurrence of cancer incidence and control. The cancer registry gathers data from various health agencies on cancer cases diagnosed or treated. For example, the Australian Cancer Database (ACD)[488] contains data about all cases of cancer diagnosed in Australia since 1982[489].

7. **Patient controlled Personal Health Record(PHR):** PHRs include patient-generated health records collected via consumer health apps, sensors and wearable devices [464]. Health data can be hosted on storage systems entirely managed by patients. For instance, patients may

collect their own blood glucose, ECG and other readings and store the data in a personal health record system they manage.

Each of the seven types of repositories for the storage of health data outlined above has different costs, security vulnerabilities, accessibility levels, usability features, and reliability track records. For example, Blockchain repositories avoid a trusted authority but are computationally very expensive. The government-run My Health Record prevents unauthorized individuals from sharing or disclosing patient data but has restricted capacity to store streamed data from sensors. Propriety Cloud eHealth repositories can provide patients with theoretically unlimited storage, but the retrieval of data can be slow.

The need to maintain privacy and confidentiality is often depicted as minimal requirements for all health data; however, in practice, health data is not equally sensitive for every patient at all times. A patient may generate her own ECG data for storage on a personal health record, allow indicators such as the ST segment to be copied to her cardiologist's record and be available to other providers through a government-operated EHR, however, rescind this when she attains a high public profile. The same patient may be compelled to accept that her provider will store her pregnancy test results but prefer that data should not be available to anyone else at all.

Health data can be thought to be disseminated among diverse agents managing storage repositories in such a way that the nominated storage medium reflects data management requirements including the quality of service, cost, volume, confidentiality, security and privacy of data that the patient desires for each chunk of his or her data. Steven[490] has taken one step toward this ideal by describing a hybrid execution model to store data defined as "sensitive" in a private Cloud and "non-sensitive" data in a public Cloud. This approach facilitates the processing of sensitive and non-sensitive data as defined by the user while preserving the user's privacy. However, this approach was not explicitly advanced for health data. Further, the communication between two kinds of Cloud platform causes long network delays and requires high bandwidth for data-intensive computation. Zhang[491] advanced a hybrid Cloud platform within the same network to address the issue.

Artificial intelligence in healthcare has made it possible to automatically diagnose health data while streaming data from medical sensors, apps and devices. An algorithm can categorize specific health data, including ECG, blood pressure, and pulse rate as normal or abnormal based on a range of conditions, and the threshold set by healthcare professionals. For example, ECG wave having RR interval, QRS complex, and QT interval within the range of [0.12 to 0.20s], [0.06-0.10s], [0.30-0.44] respectively, and R-wave is less than or equal to 0.12s is considered to be abnormal[492]. Abnormal data are usually clinically useful and important for potential research. To preserve abnormal data, Vaidehi [476] proposed a multi-agent-based health monitoring system for elderly people using Body Area Sensor Networks. Four kinds of agents named Admin, Control, Query, and Data Agent manage health records where the Data Agent classifies the medical data as normal or abnormal. Normal data is filtered out, and abnormal data is compressed to handle Big data challenges in continuous patient monitoring. However, this approach assumes a single storage medium.

Huge amounts of health data are now generated, which necessitates diverse storage options[94]. Ghamdi[493] explored different online storage systems and presented a case study for the storage of data generated from an oil company. Ghamdi identified storage-related challenges including energy consumption for operating and cooling storage, the capacity of repositories to cope with the growth of Big data, unused storage, the risk associated with data loss, downtime and backup issues of different storage mediums. NetApp platform among NAS (Network Area Storage), SAN

(Storage Area Network) and DAS (Direct Attached Storage), Cloud and Hadoop were suggested as storage mediums for Big data. A follow-up survey was conducted, which showed that NetApp facilitated data encryption, compression and solved the unused storage problem. Although they considered multiple storage mediums and assessed those against relevant criteria, oil company data, unlike health data, is relatively uniform, so no method for dynamically selecting a storage medium was proposed.

Many researchers [461–463] have developed methods for selecting suitable Cloud Service Providers (CSPs) to store consumer data, taking into account the performance and cost parameters of the CSPs. Alvarez[461,462] proposed a model that considered an application's requirements and user's priorities to choose a Cloud server among different Cloud Service Providers (CSP). They developed a mathematical model based on Linear Integer Programming with respect to storage computing cost and performance characteristics, including latency, bandwidth, and job turnaround. Yoon[463] also proposed a Linear Integer Programming model that used processing time and cost to optimally allocate datasets to distributed heterogeneous Clouds. The Cloud service with high processing power minimizes the operational time but incurs high operational costs. Conversely, the Cloud service with low processing time minimizes the operational cost but increases the processing time. As in other work cited here, this work also focused on the performance assessment of different Cloud Service Providers but did not focus on mechanisms for selecting different types of health data repository. The more general problem of how best health data can be disseminated among multiple health management systems based on data management requirements and patient preferences has still not been addressed.

Further, the Blockchain that promises security and privacy has prompted researchers to investigate it for the management of health data. However, Blockchain technologies are not an ideal solution for hosting Big health data due to its design. To address this issue, a number of researchers suggested merging traditional health databases with Blockchain-based eHealth and distributing data among them according to the user's choice and probable future data usage. For instance, Uddin et al. [89] advanced an architecture that places a software agent known as a Patient Agent that is aware of the patient's preferences, on hardware that could continuously make the storage repository decision on the basis of data sensitivity, context, significance, security and access level. However, they did not describe a feasible model for making this decision. In addition, most of the focus by Steven[490], Zhang[491] and Uddin[89] was on the development of improved cryptographic techniques to protect sensitive health data in the Cloud.

We extended these approaches by developing a model that can make the storage repository decision to select a repository amongst a range of repositories by taking into account a broader analysis of patient data beyond the "normal" or "abnormal" criteria Vaidehi[476] adopts by also taking into account other factors such as data security, privacy, and QoP (Quality of Performance) requirements. In our work, we have considered data variations in terms of sensitivity, volume, and other factors in order to direct data to one or more of the health record management systems available.

Further, the state-of-the art works have not dealt with data storage requirements but rather focused on Cloud Service Providers(CSP) selection based on diverse criteria using optimization methods. We propose a novel health data storage recommendation model to distribute health data among multiple health repositories using machine learning.

Our work involves an automated health data storage recommendation model that suggests an appropriate storage repository by considering health data sensitivity, quality of performance and patient's security and privacy preferences.

We describe relevant literature in the next section, our model after that, and evaluation trials in

the results section before concluding the paper.

5.2 Related Literature

The amount of health data has risen exponentially, with growing numbers of patients wearing bracelets and other medical IoT sensors. Each health record system cannot necessarily meet the requirements of Big data in terms of storage space, storage speed, storage structure etc. Moreover, patients are at risk of losing important medical information[494] if the correct health record system is not selected.

In some studies, the health data generated from wearable sensors, and different medical apps were manually uploaded to personal health record systems which might have delayed the response from the caregivers. To address this issue, Andy et al. [495] and Peleg et al. [496] advanced the usages of patient-generated data by uploading it to commercial blood glucose monitors. Martinez[497] developed an automated blood pressure cuff that channelled data to the HealthVault[498] hosted by Microsoft. Some research[499] has suggested filtering or compressing streamed data to fit into the electronic health record system. Hohemberger at el.[499] addressed the challenges of storing health data streamed from wearable sensors in EHR (Electronic Health Record) and proposed health data reduction policies that intended to save the heart rate of a patient in a specific range of ages.

The research in [500–503] advocated some action plans and standards to adopt an electronic health record system. However, these studies[500–503] did not develop any model to accommodate user's preferences and data storage requirements. Neil[500] urged healthcare professionals to follow three steps: assessment, planning and selection before adopting an electronic health record systems. Healthcare practitioners were recommended to recognize their requirements and affordability during the assessment process. In the planning steps, they would define their goals and identify priorities and barriers while choosing a health record system. Finally, many criteria for assessing a health record system such as time-saving, ease of use, billing, quality of service and the ability to participate in a particular insurance plan are determined in the selection phase.

Allison[501] emphasized that when choosing a specific Electronic Health Record, functional needs, troubleshooting, and optimization facilities should be taken into account. The author provided a checklist to follow before purchasing any electronic health record system. The checklist mainly covers on-site client meeting arrangements, site visiting, maintaining live workflow and others. Edmund[502] proposed ten laws to follow before choosing a specific digital health data repository: future use, volume and access time of data, backup capabilities, and privacy protections, storage costs are important factors to be considered when choosing repositories for health data[503].

Boonstra and Ross[504, 505] described several obstacles faced by medical professionals and practitioners while adopting an electronic record system. Some of these are high implementation costs and maintenance costs, legal and technical problems like system complexity, lack of support staff, low customizability. Healthcare professionals and patients are usually not incentivized for using electronic health records, which has hindered wider adoption of Electronic Health record system. Further, patients and healthcare professionals' concerns regarding privacy and security have not been addressed to the extent they expect[506].

Khan et al.[507] described the need to create a data warehouse for health data spread across a variety of sources, including clinics, hospitals, insurers, and patients. They proposed a broadly accepted conceptual and logical data warehouse model to store various types of geographically

dispersed health data. They defined two data criteria: the amount of unstructured health data and confidentiality that the data warehouse model would tackle.

Edmund[502] emphasised that medical data should be used in plaintext without filtration and compression. As the data analytics and processing method upgrade or change over time, future re-analysis and reproducibility may be possible to be carried out on the data to improve insights. Researchers can encounter difficulties in verifying potential empirical results, the validity of statistical models, and findings through studies using the derived data. However, the difficulty of maintaining raw data lies in protecting data integrity. The emerging Blockchain technology can provide a viable alternative to preserving raw data integrity. Blockchain can support the on-chain cryptographic hash code of the raw data to be maintained in a decentralised manner, which can validate the integrity of the raw data stored in off-chain.

Privacy in health informatics refers to an individual's right to monitor and control access and distribution of health data. Patients are often unable to fully control their health information, but they desire more control over their health information[506]. Individual's desire for privacy is influenced by their gender, age, the level of data sensitivity and health conditions[508]. Some research[509] indicates women are more concerned about privacy than men. Yet Kenny[508] concluded that males have greater privacy concerns regarding health data than females. Kenny has described human characteristics, behaviours and experiences as driving factors in individuals' increasing concerns about privacy. The authors verified several hypotheses through their studies. For instance, individuals are hesitant to reveal sensitive information about health. Age has a positive influence on privacy matters, with older people having more concerns about privacy. An individual with a health condition typically has less privacy concerns, as they seek to benefit from health services[508]. Rahim et al.[510] provided a conceptual model for patient privacy preferences in the healthcare system. In the model, he identified four antecedents that positively influence the patient's privacy in the healthcare environment. The antecedents described in the model include the needs for exchanging health data, the patient's faith in the EMR, the ease of access control in the EMR and patient's security awareness.

Many studies[97,511–514] identified a wide range of parameters for evaluating Cloud services and proposed some guidelines that should be followed when choosing health records. We reviewed the following literature that advanced standards for assessing health record systems in order to design our proposed model.

Chang et al.[511] developed an objective mathematical framework for maximizing benefits with a given budget and cost to minimize the likelihood of CSP failure and improve availability. DP(Dynamic Programming) was used to select the best CSPs. The method maximises the number of data blocks that survive when certain CSPs fail, or are subject to a fixed budget. Rehman[512] put forward a framework for tracking the performance of CSP through feedback from users. Qu[513] introduced a CSP selection process based on user feedback that includes four components; Cloud Selection Service, Benchmark Testing Service, User Feedback Management Service and Aggregation Evaluation Service. Qu defined the criterion for choosing CSP as subjective or objective. Cloud consumers give ratings as subjective criteria to the system, and third party trust supplies the system with measurable CSP performance as objective criteria. A simple Additive Fuzzy System which aggregates subjective and objective criteria were used to rank the available CSPs.

Lee[97] suggested a hybrid multi-criteria decision-making model for CSP selection in which, initially, decision-making factors were defined using a Balanced Scorecard (BSC) methodology. Secondly, critical decision criteria were extracted using the Fuzzy Delphi (FDM) process and thirdly, weight allocation for each decision-making criterion and CSP selection was carried out

using FAHP(Fuzzy Analytic Hierarchical Process).

Halabi[514] hierarchically identified a set of criteria for evaluating the security of CSPs, where security was subjectively and objectively evaluated using the Analytic Hierarchical Process (AHP). In order to comply with a CIA(Confidentiality, Integrity and Availability), Halabi[515] has also introduced a broker-based system that will fulfil the Service Level Agreement. They developed a CIA-based optimization function to identify CSPs with minimal user frustration for CIA. Halabi[516] addressed online Cloud services allocations in view of global safety satisfaction. A linear optimization technique is used to formalise the resource allocation problem in relation to global security requirements. The linear optimisation problem formulated was solved using a genetic algorithm.

Patient-centered health data with structural heterogeneity are produced at a particularly high rate, and high magnitude so needs to be stored and processed rapidly. Precision is crucial to extract useful insights from health data, but some sources generate vague and inaccurate data. Nonetheless, a distributed data management system can resolve these issues to some degree.[484].

The studies discussed above have explored diverse Cloud storage mediums. However, these studies did not develop machine learning-based mechanisms to meet user's preferences and data features and also did not design the selection of repositories considering various health data storage systems and data properties. Our approach for facilitating distributed health data management is outlined in the next section.

5.3 The Health Repositories Recommendation Model for Health Data

The storage recommendation system presented here assumes the patient is in control of the storage decision along the lines advocated by the "Gimme me my dam data"[517] movement. In many jurisdictions, health information generated by healthcare providers is owned and controlled by a healthcare provider. However, as consumer health movements increase in popularity and increasingly patients generate their own data, storage decision's are assumed to become more pressing for patients. Further, as the quantum of streamed data increases, storage decisions must be made so frequently that manual consultation with the patient becomes cumbersome, and an automated process is required.

The process advanced here maps information about the storage requirements a patient has for a block of data to the storage features of repositories managed by diverse agents. However, a patient's data storage requirements vary enormously and cannot necessarily be pre-specified to cover all future patient contexts. This is managed by having a mapping manually specified by experts as a training set for a machine learning classifier to learn to generalize to a mapping that covers a wider set of patient contexts. Figure 5.2 and 5.3 show the overall approach developed here, explained in detail below. First, we describe a set of variables or features characterising the requirements for storing a chunk of data- *the data storage requirements* which is illustrated in Phase-1 of Figure 5.2.

Some of the attribute's values for data storage requirements are declared to be numerical (range between 1 and 5) and some are categorical. Secondly, a dataset having these attributes or features is constructed where each instance reflects the specifications needed for storing a particular chunk of data (This constitutes Phase-2 of the model shown in Figure 5.2).

Next, the features that reflect characteristics of storage repositories called the *Health Repositories Evaluation Criteria* are calculated by adding the rating provided by an expertise group. This

is presented in Phase-2 of Figure 5.2. Throughout this scenario, we are ranking five storage repositories against four standards. In Phase-3 of Figure 5.2, ultimately, statistical correlation, clinical heuristic rules (those rules can be created by the medical professionals or patients themselves), and user preferences are used to decide the class labeling for each instance in the dataset. The experts or users may, in a real situation, allocate a storage repository (class label) to an instance that will be encoded using heuristic rules. The correlation coefficient is used to infer the class label of those instances for which a user's preferences or heuristic rules are not exactly matched.

In Phase-4 of Figure 5.2, a machine learning classifier trained with the sample dataset containing user and expert expectations can, therefore, generalise the mapping of data requirements to health repositories. The storage recommendation framework shown in Figure 5.3 comprises two parts: the selection of data storage requirements and assessment standards for health repositories, and Machine Learning. Each component is described below.

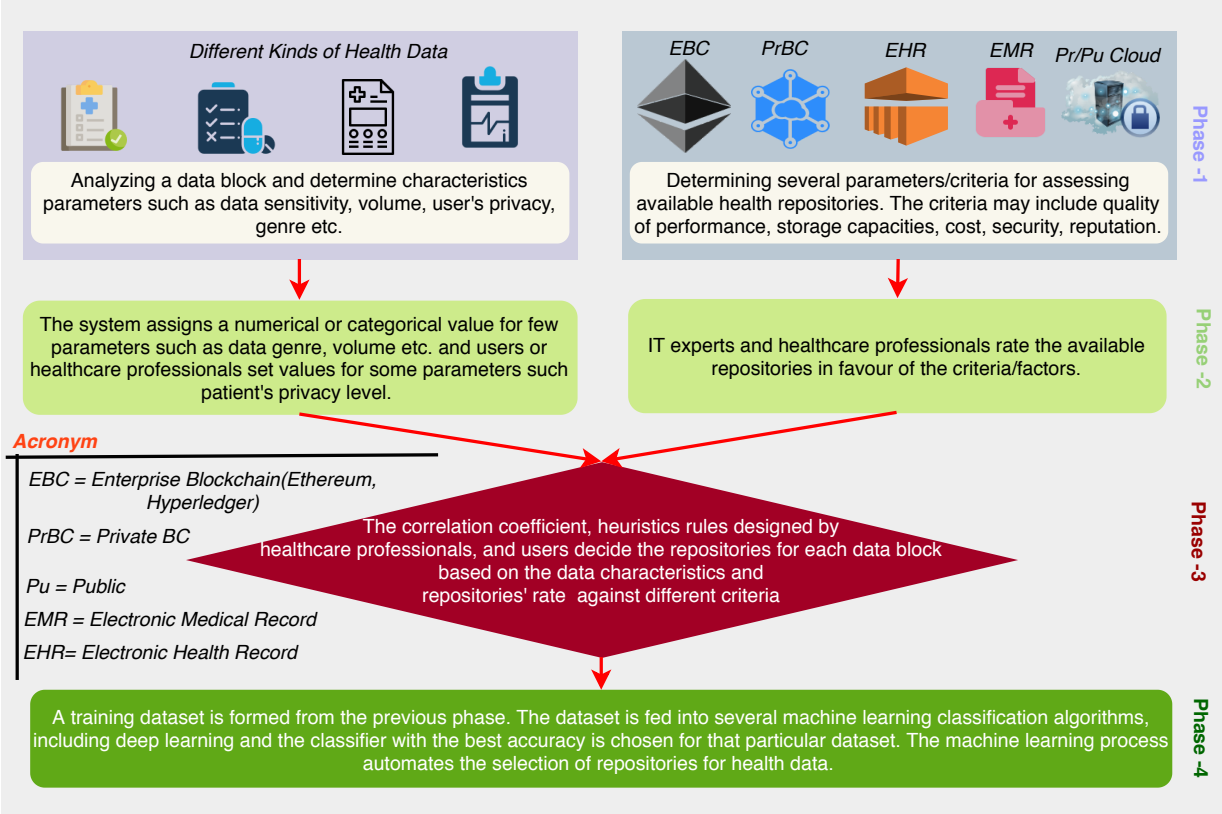


Figure 5.2: The high level view of the proposed recommendation model

5.3.1 Data storage requirements and health repositories assessment standards selection

The upper part of the framework in Figure 5.3 includes features that reflect characteristics of the data to be stored called *Data Storage Requirements*, and features that reflect characteristics of storage repositories called the *Health Repositories Evaluation Criteria* and an association analysis between the two sets of features.

5.3.1.1 Data Storage Requirements

The requirements considered relevant for deciding which repository should best be used for a chunk of data have been selected from the literature and include sensitivity, volume, medical care context and patient demographic data:

- **Sensitivity:** Although all health-related data should be prevented from unauthorized access, some data can be regarded to be more sensitive to breaches than other data. The level of data sensitivity can be expected to vary from individual to individual, depending on their personal preferences and contexts. For example, data concerning a person's sexual orientation may be highly sensitive for one person in one context compared with another person in the same or different context. To illustrate, an ECG trace at one point may need to be kept extremely secure against unauthorized access for one patient but less so at another point in time.

- **Volume:** Is the data block a single small block as in a test result or is the data streaming forming huge datasets such as continuous streams including ECG, blood pressure, temperature, and oxygen level? This latter dataset requires health storage repositories that can support virtually unlimited storage, whereas static reports, medical diagnoses and medication summaries are occasionally generated and do not need a storage medium with high capacity.

- **Medical Care Context:** Although many contexts patients find themselves in can be identified, a small number of contexts can be identified at a coarse-grained level. For this work, four contexts were considered sufficient to describe common medical care contexts: a palliative care context, emergency context, chronically ill context or non-chronic disease context. Medical care contexts can also be expected to vary from country to country. For example, in Australia, medical contexts might include front line care (GP), hospital care, emergency care, specialist care, allied care, elderly care and palliative care. Different care contexts can be served by storage repositories to different extents. For example, having health data stored in EMR managed by healthcare providers is more desirable during emergency or life-threatening contexts because it can be retrieved quickly.

- **Patient Demographics:** Data such as socio-economic status, profession, education and nationality can play a significant role in the selection of a storage medium. For instance, storage cost may be particularly important for a person on a low income, whereas confidentiality may be very important for a person with a high public profile.

5.3.1.2 Health Repositories Evaluation Criteria

Table 5.1 and 5.2 illustrate features that distinguish four of the organizations that manage health data repositories described above. While many factors distinguish one manager from another, we limit our focus to four: security and privacy, performance quality, capacity and cost. Each of the four main criteria has sub-criteria. Criteria related to the performance of a repository, such as downloading or uploading speed, data availability, and maintenance services are clustered as *Quality of Performance* criteria. Likewise, criteria related to security and privacy, such as the capabilities of preserving confidentiality, data integrity, and resistance to cyberattacks are listed as *Security and Privacy*. Figure 5.4 shows criteria and sub-criteria against which health repositories are assessed.

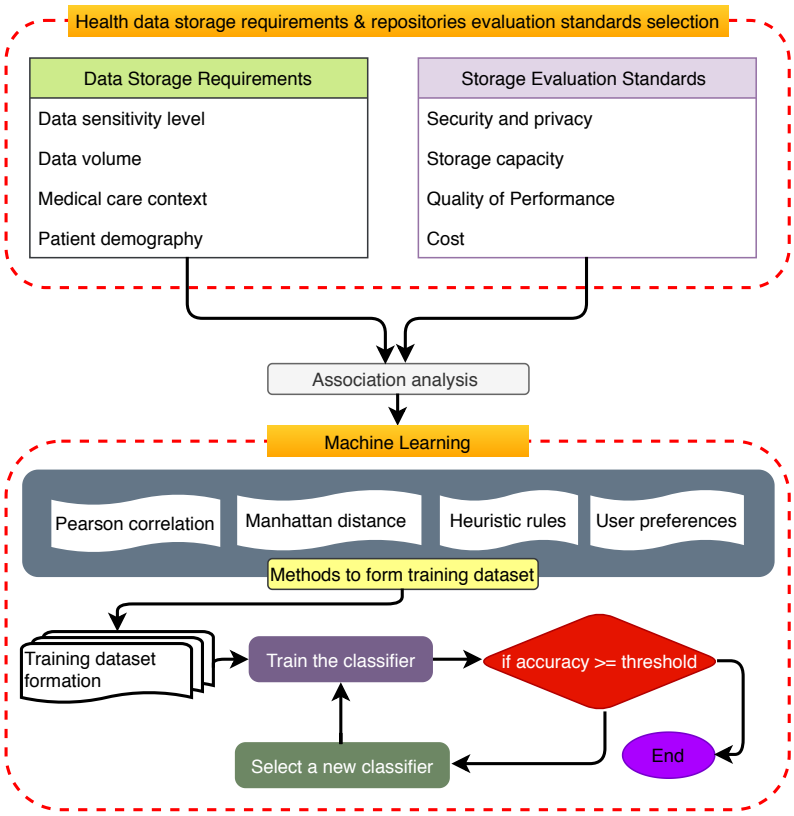


Figure 5.3: The health data storage recommendation systems

- **Security and privacy:** This includes **confidentiality** that represents the capacity for a storage medium to protect patient's data against inappropriate disclosure or tampering by the insider or outsider attackers. Some storage repositories have better cyberattack defenses than others. Additionally, some storage repositories can keep data accessible at all times or deliver data upon request, including unexpected disruptions like hardware failures, cyberattacks or natural disasters better than others. For some repositories, only the receiver and sender are involved in processing patient data, whereas others involve third-parties. Some repositories enable the patient to control access to his or her data to a greater extent than others(access control)

- **Quality of Performance** criteria includes **processing speed** that indicates the time of uploading, downloading and processing patient health data, **interoperability** refers to the ability of a storage medium to exchange data among different kinds of systems and software, and **data transparency** refers to the capability of a storage medium to ensure correctness, the legitimacy of the data source and the capacity to easily access and use data irrespective of source and location. The **storage organization's reputation** represents the past history of the storage repository manager's ratings from bodies such as investors, customers, suppliers, employees, regulators, politicians, non-governmental organizations for its service.

- **Storage capacity** indicates the capacity of a storage repository to backup and archive data, and **durability** refers to the capacity for a repository to protect patient's health-related data from bit rot, degradation, and other long term corruptions.

- **Cost** involves deployment, and maintenance that indicates the action taken by a storage medium to retain or restore its service or machine, equipment, and service.

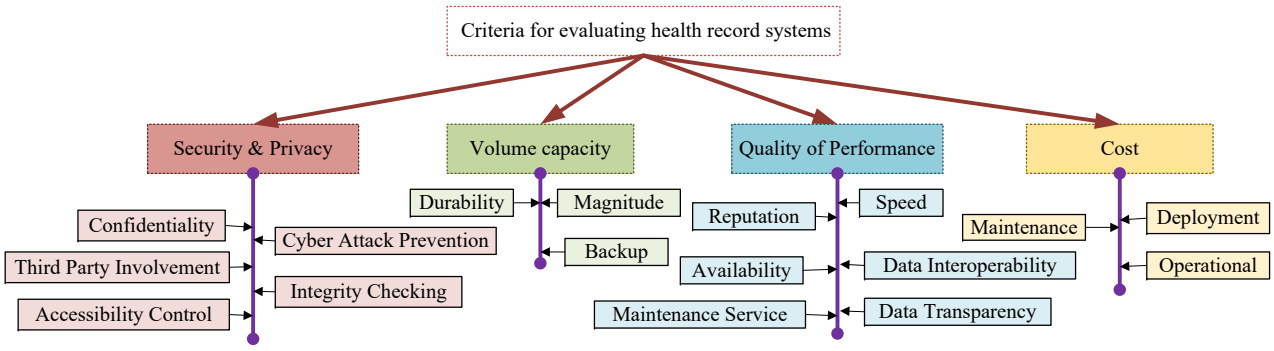


Figure 5.4: The hierarchical representations of health repositories evaluating standards

Table 5.1 and 5.2 describe the strengths and weaknesses of four storage repositories against the four major criteria: security and privacy, quality of performance and cost. Table 5.3 presents the assessment of five health data repositories against the sub-criteria under four major criteria. Values range from [1 to 5] for each feature. The ratings derive from the three of the authors' own judgements, as IT experts. Future research is planned to source the ratings from a wider group of IT experts and healthcare professionals. The single rating for criteria is calculated by averaging the ratings provided by the three authors. The rating in favor of a criterion for a health data storage repository is estimated according to equation (5.1).

$$r_{i,j} = \frac{\sum_{k=1}^{n} x_k}{n} \text{ where } x_k = \frac{\sum_{c=1}^{m} r_c}{m} \tag{5.1}$$

$r_{i,j}$ indicates rate against a criterion i for a storage medium j. x_k indicates a rate given by an expert against the criterion i for the storage medium j. r_c represents rating given by an expert against sub-criteria. The radar graph depicted in Figure 5.5 visualizes the strength of five health repositories with respect to the four criteria.

5.3.1.3 The association between data features and repository evaluation standards

The proposed method aims to transfer medical data, particularly patient-generated health data to one of the health record systems that appropriately reflect the data requirements or user's preferences. Health data requirements outlined above are associated with storage evaluation criteria in a one to many relation where some associations are strong, and some are weakly related. Figure 5.6 shows the relationship between data storage requirements and storage evaluation criteria. The data features have interrelationships and effect one another. For example, a data block considered highly confidential may be submitted in plaintext format to a health record system for rapid processing. At the same time, a patient's demographic features(such as high social status or public profile) can make relatively low confidential data highly sensitive. Demographic data, such as education or technical experience, is likely to positively influence patient privacy concerns. So he or she can choose a particular storage repository that protects health data confidentiality.

Table 5.1: The Strength and weakness of health repositories against Criteria

Criteria for evaluating health repositories	Government EHR	Blockchain EHR
Security and Privacy	Government employees may access health data without patient's knowledge. EHR realizes legal compliance constraints enshrined in legislation[427]. EHR implementations are subject to rigorous audit process, minimising the risk of data manipulation.	Blockchain EHR is a patient-driven data management technology that prevents unauthorized access to records. Blockchain EHR can anonymously process health records and guarantee information integrity by copying the entire ledger to multiple entities. However, a patient's privacy is breached if attackers can discover the data owner through content analysis [367]. Although Blockchain EHR withstands major cybersecurity attacks such as Denial of Service (DoS), Ransomware and single point of failure, it is susceptible to protocol related attacks such as a long-range attack, and mining attacks known as 51% attacks.
Storage Capacity	Government EHR is a scalable storage management system but not suitable for streamed data. Although EHR facilitates an extensive archive of patient medical history with a high level of security, uploading streamed data to EHR is impracticable due to a large amount of data that needs to be stored over the time [499].	Blockchain does not provide scalable storage facilities for mining Big health data on-chain as the record is required to be replicated in every participant [161]. However, off-chain data management in the Blockchain can meet this challenge.
Quality of Performance	EHR maintains standardized and uninterrupted coordination services promptly.	Blockchain EHR can support cross border sharing of health data while preserving confidentiality and integrity. User can access data from various points. However, slow processing and access to health data [426] due to limited scalability, legal and political compliance issues[518] can impact the quality of care.
Cost	Government EHR requires high implementation, maintenance and administrative costs that many national governments might not afford. However, government management of EHR maximizes cost-effectiveness and quality of care for the patient. [425]	Blockchain EHR alleviates many service costs, including employee wages, a legal fee but users have to contribute computational resources.

EHR = Electronic Health Record, Blockchain EHR = Blockchain based Electronic Health Record

Table 5.2: The strength and weakness of health record systems

Criteria for evaluating health repositories	Proprietary eHealth Cloud Provider	Healthcare Provider EMR
Security and Privacy	Patient's identifier and health data is accessible by Cloud administrators[519] that threatens patient's privacy. It cannot guarantee the integrity of health data due to third parties' involvement in processing and providing storage[83] [520]. Further, Cloud database is prone to many cyberattacks, including data breaches, prefix hijacking[519], spoofing identity, trust management and non-repudiation among servers. However, top Cloud providers such as Microsoft, Amazon web Service safeguard customer's data from malicious attacks and facilitate the availability and access to data across multiple organizations located worldwide.	Insiders such as healthcare professionals, and support staff are associated with over half of recent health data breaches[400] in EMRs. EMRs are defenceless against different cyberattacks, including DoS, ransom, and single point of failure. Risks of information leakage during data dissemination. Laws and regulations bar the rapid sharing[424] of EMR data with other organizations from different countries. However, an organization managing EMR provides its healthcare professionals with instant access to each patient's history, allowing the practice to track patient history and identify patients who are due for visits, tests or screenings.
Storage Capacity	Cloud virtually provides flexible and scalable storage to mine, manipulate and analyze large health datasets[518]. However, Cloud servers may occasionally encounter operational failure causing unavailability of data.	EMR is built with limited storage capacity that accommodates health information from a single institution but not appropriate for continuously streamed data.
Quality of Performance	Cloud causes some delays in handling massive numbers of entities depending on the quality of internet connections. However, Cloud facilitates seamless, and timely transmission and sharing [521] health data worldwide.	EMR system enables healthcare professionals to exercise consent exception in an emergency (insufficient time to pursue informed consent from a patient) which improves the quality of care. However, the EMR system provides inadequate interoperability while sharing health data across different health organizations due to their diverse security and access policies[521].
Cost	Cloud offers a cost-efficient, more effortless scalable environment for storage and deployment of applications.	Most health organizations prefer on-premise storage which costs higher than Cloud-based storage options.

EMR = Electronic Medical Record

Table 5.3: Rating five health repositories against four criteria

Evaluation Criteria	Sub Criteria	BC EHR	Cloud eHealth	EMR	PHR	EHR
Security & Privacy	To what extent can the storage repository ensure data integrity?	4.65	2.85	2.90	3.40	2.85
	To what extent is the storage repository available 24/7?					
	To what extent can a third party access data?					
	To what extent can the storage repository withstand Ransomware, DoS, Insider Attacks?					
Storage Capacity	To what extent can the repository support storage for Big data?	1.67	4.42	3.1	1.50	2.77
	To what extent can the repository facilitate processing of Big data?					
	To what extent can the repository facilitate storage for continuously streamed data?					
QoP	How fast can data uploading be?	2.00	3.67	3.52	3.17	3.52
	How fast can data retrieval be?					
	How fast can data processing be?					
Cost	How low is deployment cost?	3.83	4.05	3.44	1.73	4.40
	How low are maintenance costs ?					
	How low are service costs?					

BC EHR = Blockchain Electronic Health Record, EMR = Electronic Medical Record, PHR = Personal Health Record, QoP = Quality of Performance

Table 5.4: Relation between data storage requirements and repository evaluation criteria

Data Requirements	Remarks	Storage Evaluation Criteria
Data Sensitivity	In general, all medical data is not labeled with the same level of sensitivity. For instance, ECG data for a person with a high public profile may be sought by so many commentators that the level of security required is extreme. The data sensitivity is intimately associated with the security and privacy capacity of a storage repository.	Security and Privacy
Data Volume	Large volume of data should be channeled to a storage medium with high capacity, and low volume of data can be stored in a storage medium with lower capacity. So, data volume is linked to the storage capacity of a repository.	Storage Capacity
Care Context	Access to health data might tolerate a certain amount of delay depending on the types of care. For instance, the delay can be tolerated in normal care setting but not in an emergency setting. So, different levels of QoP need to be ensured on the basis of care status.	QoP (Quality of Performance)
Socio-economic	A patient's demographic profile plays a role in deciding how much privacy, security and quality of performance a patient requires when selecting a health repository. In developing countries, the socio-economic status of a patient may be closely linked to the costs associated with a repository	Cost

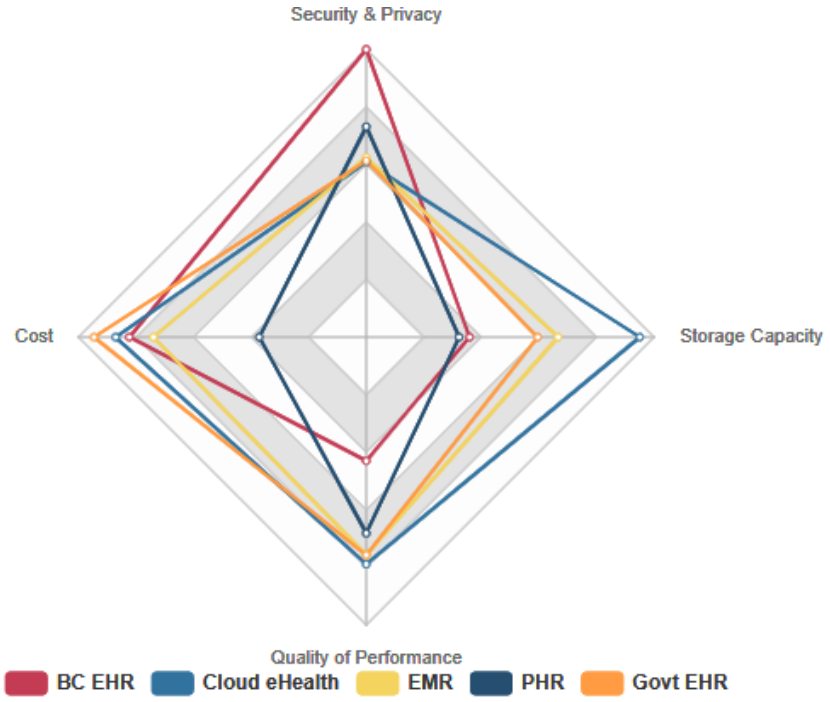


Figure 5.5: The strength of five health repositories in favor of four criteria

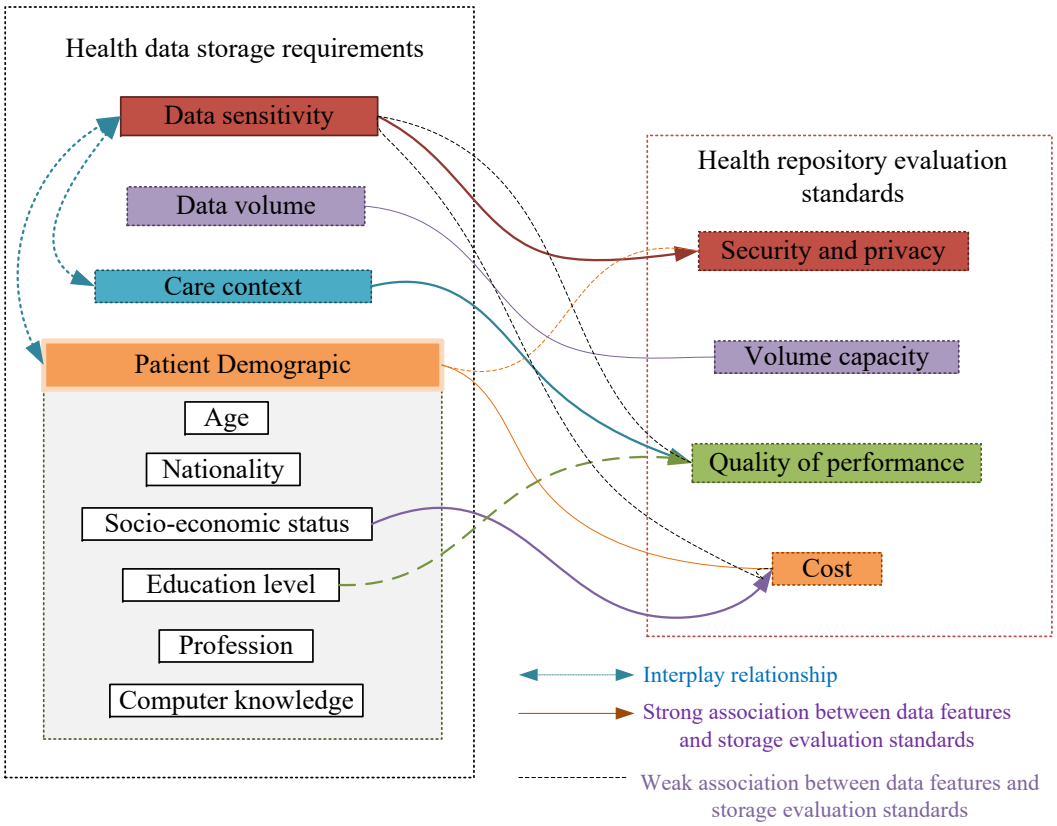


Figure 5.6: The mapping between data storage requirements and storage medium evaluation criteria

5.3.2 Machine learning

This section describes how a training dataset that represents the mapping of the different data blocks to various health repositories are created for the machine learning algorithms. The system adopts supervised learning for dynamically suggesting health repositories for a particular data block. For this reason, we need to generate a training dataset with the label for each instance of the dataset.

5.3.2.1 Mapping between health data block and health repositories

We have taken into account a few methods to determine the class label (health repository) for each entity in the dataset. The approach includes correlation coefficient analysis, distance measurement, heuristic rules designed by healthcare professionals, and user preferences.

- *Mapping using correlation coefficient:* We specify several features for each data block to be assigned to a health repository. Some of these features are directly related to the data block, and some features are associated with the patient. The features might include the level of sensitivity, the magnitude or volume of data, data type, medical care context, and patient demographic information(nationality, profession, education and socio-economic status and income level).

  Firstly, four features named data sensitivity, volume, medical care context and consumer's income level have a linear correlation with four attributes of health repositories: security and privacy, storage capacity, quality of performance and costs associated with adopting a health record system. The association between data features and the criteria of the health repository is explained in Table 5.4.

  Each data feature shown in Table 5.4 is assigned a value in the range [1 to 5]. For example, a specific health data block assigned to "higher confidential" has value 5 for the sensitivity feature, and medium one has value 4 for that attribute. Similarly, a data block with high magnitude has value 5 for the data volume feature and so on.

  Pearson correlation coefficient is calculated to label an instance provided that other features do not have an impact on deciding the health repositories.

  The Pearson correlation coefficient is presented in equation (5.2). We calculate the correlation coefficient between four features of a data block and four evaluation criteria of all health repositories. The repository with the highest Pearson coefficient with respect to features of a data block was considered best suited for that data block.

$$r_i = \frac{\sum_{j=1}^{m}(x_j - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{j=1}^{m}(x_j - \bar{x})^2}\sqrt{\sum_{j=1}^{m}(y_j - \bar{y})^2}} \tag{5.2}$$

  Assuming that r_1, r_2, r_3, r_4 and r_5 are calculated between the set of data storage requirements(D) and the evaluation criteria of EHR(S_1), PHR(S_2), Cloud eHealth(S_3), Blockchain(S_4) and EMR(S_5) respectively.

  The recommended storage(S_i) for a particular instance of the dataset D is estimated using equation (5.3):

$$S_i = \max(r_1, r_2, \ldots, r_n) \tag{5.3}$$

where $i = 1, 2, \ldots, n$ and $j = 1, \ldots, m$). n is number of storage mediums and m is the number of criteria.

However, if any instance with identical values for all the features appears in the dataset, the Pearson correlation coefficient cannot be calculated to discover the best-suited repository for that instance. In such cases, the Euclidean or Manhattan distance between data storage requirements and the criteria of all repositories is calculated to determine the best-fitted repositories for storing the data block.

Assuming that, the recommended repository(S_i) for a particular instance I that has identical value for all the features can be found using equation (5.4)

$$r_i = \min(\sum_{i=j}^{m} |x_{i,j}, -y_{i,j}|) \tag{5.4}$$

$$S_i = \min(r_1, r_2, \ldots, r_n) \tag{5.5}$$

where $i = 1, 2, \ldots, n$ and $j = 1, \ldots, m$

- *Mapping using experts' knowledge:* Secondly, healthcare professionals' decision, user's preferences and other features such as normal or abnormal patterns, patient profile status and other demographic factors can dominate in selecting an appropriate health data storage repository. For instance, unusual heart patterns in cardiovascular patients are likely to be clinically useful and should be stored in such a repository that enables rapid access by healthcare professionals. Data that is within normal ranges can often be stored in a low secured or inexpensive storage repository because it is unlikely to be of interest to future health care professionals, though may have minimal utility for future health research. Additionally, selection of health repositories also relies on the data block genre. For instance, in many countries such as Australia, USA, Europe, data related to a cancer diagnosis is uploaded to a cancer registry database.

  Heuristic rules can best address the contexts discussed above. Patient specific heuristic rules can enable high-level user preferences (healthcare professionals) to be easily specified. The heuristic rules are set to take precedence over the correlation analysis method for nominating the most appropriate storage repository for a data block. Sample rules representing the authors' preferences are as follows.

  1. **if** *Data* reflects *Normal patterns* **and** *Data volumes* are *high* , **then** *Storage medium* is *Cloud eHealth*
  2. **if** *Data reflects Normal patterns* **and** *Data volume* is *small* , **then** *Storage medium* is *PHR*
  3. **if** *Data reflects abnormal patterns* , **then** *Storage medium* is *EMR*
  4. **if** *Public profile* is *high* **and** *Care context* is *normal* , **then** *Storage medium* is *Blockchain eHealth*
  5. **if** *Public profile* is *high* **and** *Care context* is *emergency* , **then** *Storage medium* is *EMR*
  6. **if** *Data genre* is *cancer*, **then** *Send a copy of data to the Cancer registry*

- *Mapping decided by users:* The decisions regarding how data is to be disseminated among multiple storage managers should be made in accordance with a user's preference. Different users may have quite different choices regarding privacy, and the preferences may change depending on a diverse range of contexts[522]. The patient is expected to choose his or her health record systems depending on his or her health condition, demographic data (age, nationality), social profile or status, data type, sensitivity and significance of data. For example, one user might give preference to having their vital signs data stored on healthcare providers storage for rapid access in an emergency setting but not in other contexts. Another patient may be a government employee who is reluctant to have their psychiatric record on a government-managed EHR. A patient with a low public profile may not need a level of high security for his or her ECG data. In contrast, a celebrity with a high profile may prefer his or her ECG data to be stored solely in a Blockchain. Most people may reveal their blood group, whereas individuals with a high public profile may be more reluctant to do so. Further, an individual's preference regarding the level of privacy and security may change over time. A young person may desire higher security and privacy than a palliative patient. The present study aims to incorporate user preferences regarding health record systems.

5.3.2.2 Generating synthetic data

A training dataset is constructed using the above mentioned Pearson correlation coefficient, Manhattan distance and heuristic rules to train a classifier. Table 5.5 represents a sample training set where the data block features include sensitivity level, data volume (DV), medical care context, and socio-economic status (SES), public profile (PP), data type. These features' value range from 1 to 5. The class label for the first and second instance is fixed by using the Pearson correlation. Public profile and data type for these two instances are overridden because public profile value is **low** and data type is **normal**. In the fourth instance, data type is **abnormal**, which overrides the role of other features and the health data block is directed to healthcare professional providing Electronic Medical Record for having rapid health services.

Table 5.5: The sample training dataset for machine learning

Data block	Sensitivity	DV	Care context	SES	PP	Data type	Storage medium
block1	4	2	1	4	low	normal	EMR
block2	1	4	5	4	low	normal	Cloud eHealth
block3	1	1	1	1	low	normal	BC_EHR
block4	2	2	2	2	low	abnormal	EMR
...	...	...	...	...	...	...	

DV = Data volume, SES = Socio-economic status, PP= Public profile

We selected a supervised machine learning model over a rule-based expert system for suggesting health repositories for the following reasons. Large numbers of rules are required to be generated as features of data storage requirements increase. The machine learning algorithm can learn user's preferences about healthcare record systems under a diverse range of contexts. The supervised learner is trained with a pre-defined preference data to channel health data to available

health repositories automatically. User's preferences cannot be encoded using generic rules because the user's preferences about health repositories are subjective and vary from individual to individual.

Rule-based AI(Artificial Intelligence) can infer conclusions in clearly defined and bounded situations. In contrast, ML (Machine Learning) can generalize conclusions along multiple dimensions, which can model more sophisticated behaviours than a sample matching. Selection of a particular storage repository for health data is stochastic. The healthcare professionals or users may prefer a health storage system under specific data storage requirements which might be challenging to represent using rules. A machine learning algorithm can produce the best-fitted output for the cases mentioned above.

5.3.2.3 Train classifiers

We formed a training dataset using the methods described in the *machine learning* section above. A sample of such training data is illustrated in Table 5.5. We assume that we have different data blocks that can contain discharge summaries, pathological results, psychiatric evaluations, and medical images or data continuously streamed from wearable sensors. In this experiment, our target is to investigate how well the classifiers learn the data distribution rules.

The four separate training datasets have size 500, 1000, 1500, and 2000 instances, respectively. The four datasets have been fed into five different classifiers to study the feasibility of a machine learning algorithm in selecting an appropriate storage medium. Five different classifiers trained here are Multilayered Perceptions (MLP), Random Forest (RF), J48, K-nearest neighbor (IBK) and Naive Bayes (NB). The classifiers are trained using a variable size of the synthetic dataset in Weka ToolKits[523] and evaluated in terms of the following metrics.

- Confusion matrix[524] shown in Table 5.6, also called contingency table, describes the results of classification. The upper left corner True positive is the number of entities being classified as true positive while those were true. The lower right cell False-positive represents the number of samples being classified as false negative while they were false. False-negative indicates the number of entities being classified as true although those were false. False-positive represents the number of entities being classified as true, although those were true.

  $$\text{accuracy} = \frac{\sum \text{True positive} + \sum \text{True negative}}{\sum \text{total samples}}$$

Table 5.6: The confusion matrix

	Condition positive	Condition negative
Predicted condition positive	True positive	False Negative
Predicted condition negative	False positive	True negative

$$\text{Precision} = \frac{\sum \text{True Positive}}{\sum \text{Predicted condition positive}}$$

$$\text{Recall} = \frac{\sum \text{True positive}}{\sum \text{condition positive}}$$

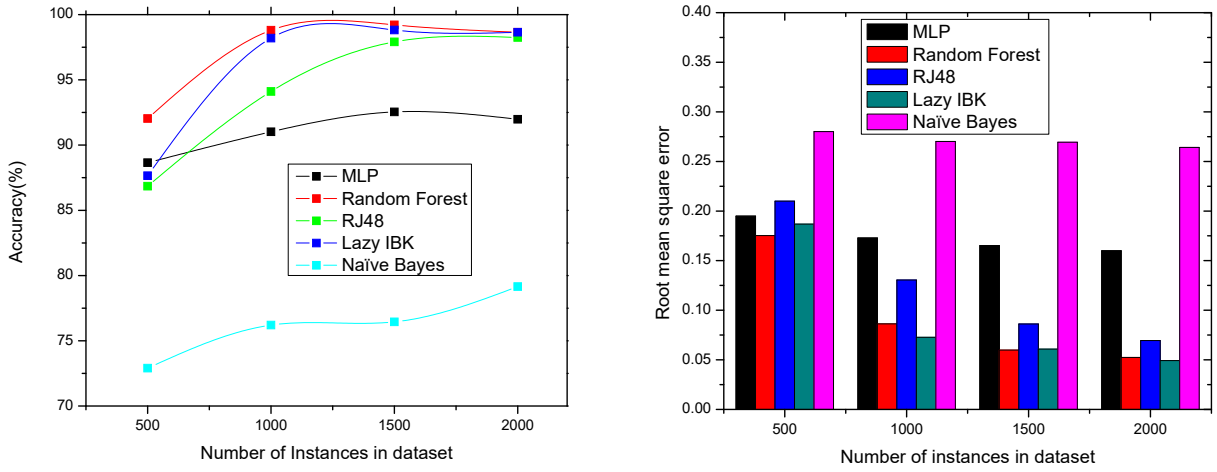
- MSE is measured by taking the square average of the difference between the data's original and predicted values. RMSE (Root mean square error) is the normal variance of the errors

that occur while predicting on a dataset. This tests about how far from the actual output the forecasts were. RMSE is defined in mathematical terms as follows.

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(\text{actual values} - \text{predicted values})^2}$$

- Receiver The Operating Characteristic Curve (or ROC Curve) is a plot of the true positive rate against the false-positive rate for the various possible diagnostic test cutpoints. ROC reveals the trade-off between sensitivity and specificity (a decrease in specificity will follow any rise in sensitivity). The more the curve follows the left border and the more closely the curve follows the top border of the ROC space, the more accurate the test.

The accuracy and ROC curve for both 10-fold cross-validation and percentage split are illustrated in Figures 5.7, 5.8, 5.9 and 5.10 respectively. The graph depicted in Figure 5.7(a) shows that Random Forest and Lazy IBK (K-nearest neighbor) classifiers offer higher accuracy with an increasing number of instances of the dataset in 10-fold cross-validation method. All the classifiers showed higher accuracy for the dataset having 1500 tuples because this dataset contains a balanced ratio of every class. Random Forest shows the highest accuracy of 99.21% and the next best classifier for this dataset is IBk that showed an accuracy of 98.82%. In contrast, all the classifiers with the dataset that has 2000 tuples showed a slightly lower accuracy largely because the dataset is imbalanced. The root mean square errors for 10-fold cross-validation is presented in Figure 5.7 (b) where Random Forest and IBK are showing less RMSE in comparison to other classifiers.



(a) Classifier's accuracy

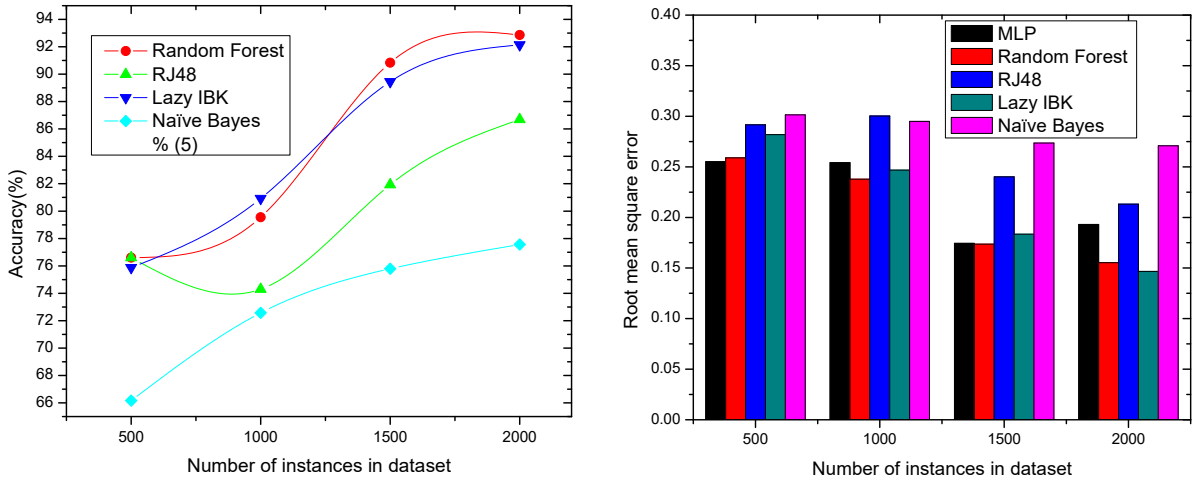
(b) Classifier's Root Mean Square Errors

Figure 5.7: 10-fold cross validation

On the other hand, the percentage split results depicted in Figure 5.8 present comparatively lower accuracy than 10-fold cross-validation. In percentage split, the dataset is partitioned into a training set(80%) and test set(20%) and classifier are trained once then all the classifiers showed low accuracy and high RMSE depicted in Figure 5.8(b).

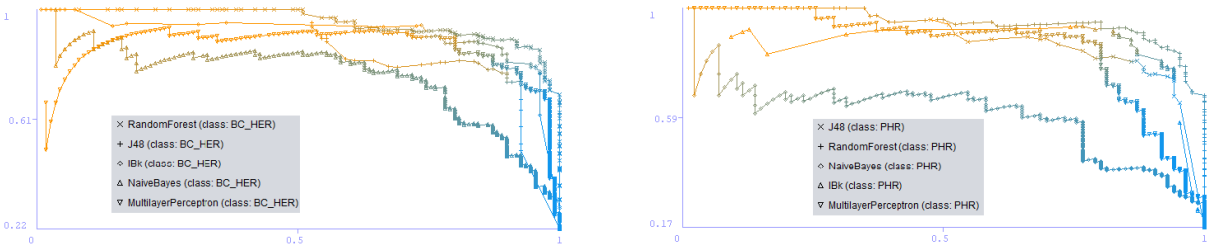
The graph depicted in Figure 5.9 and 5.10 shows the Recall vs Precision and ROC curve for different classes in 10-fold cross validation method.

Deep learning is a subset of machine learning in artificial intelligence (AI). The deep learning networks are capable of learning unsupervised data that is unstructured or unlabelled. The datasets for rapidly recommending health repositories can be unstructured and unlabelled in a real situation. So, we adopted a deep learning approach for investigating the accuracy for our synthetic datasets.



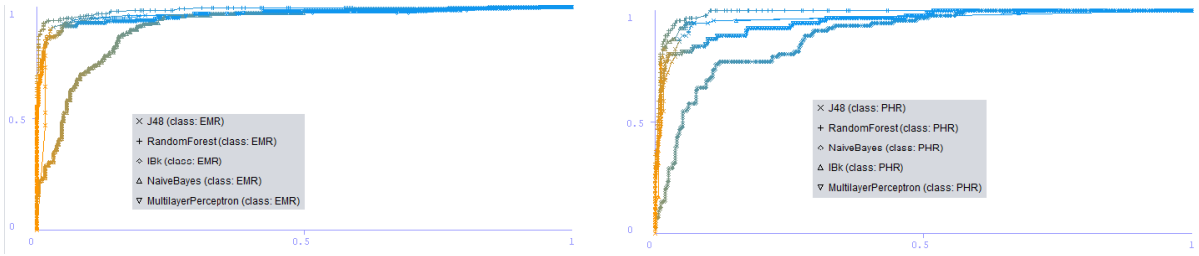
(a) Classifier's accuracy (b) Classifier's Root Mean Square Errors

Figure 5.8: Percentage split(20% testset from training dataset)



(a) Blockchain Electronic Health Record (b) Personal Health Record

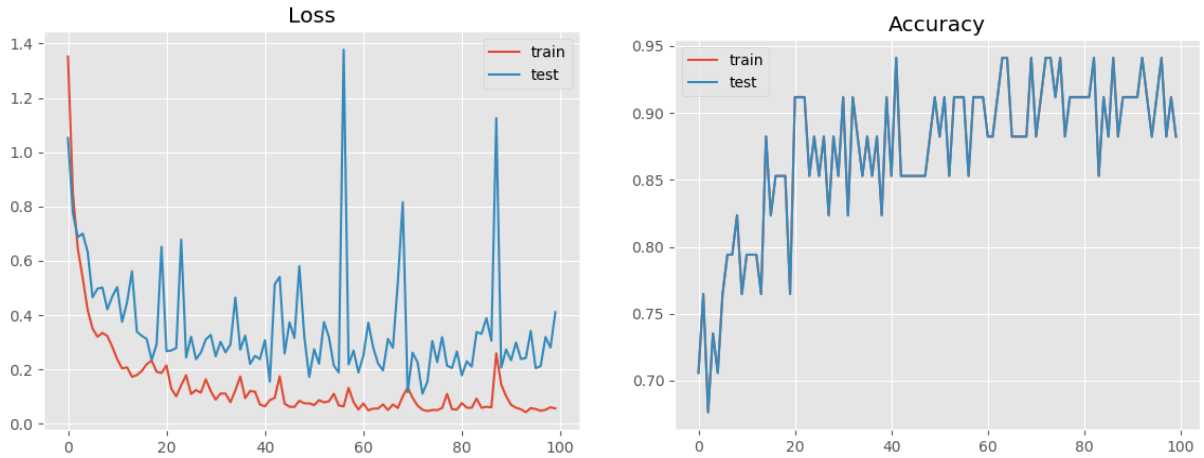
Figure 5.9: Recall vs Precision



(a) Electronic Medical Record (b) Personal Health Record

Figure 5.10: ROC curve

The synthetic dataset is fed into a deep learning model, and the model shows around 89% accuracy. The deep learning approach is modelled using Python with Keras framework. The data is based on seven input diameters with multiple classes. The model has three hidden layers where the first hidden layer has 100 output nodes that take input from 7 input diameters, and the last hidden layer has five output nodes. The model is trained using 100 epochs, and the batch size is set 8. The Confusion matrix and the accuracy in terms of different metrics are presented in Table 5.7. Figure 5.11 shows the training loss and accuracy of the sample dataset where X-axis indicates the number of epochs and Y-axis indicates loss or accuracy.



(a) Training loss

(b) Training accuracy

Figure 5.11: Result of deep learning model

Table 5.7: Accuracy of the deep learning model

	Precision	Sensitivity or recall	f1-score
Cloud eHealth	0.93	1.00	0.96
PHR	1.00	1.00	0.96
EHR	1.00	0.93	0.96
EMR	0.85	0.96	0.90
Blockchain eHealth	1.00	0.92	0.96
accuracy			0.89

The accuracy level of the classifier for the dataset demonstrates the feasibility of using machine learning or deep learning to learn the mapping between health storage mediums and a health data block.

With the rapid growth in the volume of health data that needs to be stored and accessed globally, this machine learning model could be an essential tool for improving the storage and access arrangements for the future. This method has the potential to enhance the consumer's ability to manage their health data storage and access, while also ensuring data stores are manageable from a size perspective. The ML model can assist with determining the most 'fit for purpose' storage solution for different data assets.

5.4 Adoption of new health repositories

In this paper, seven different health record systems are described as potential repositories for patient-generated health data. Five of the most prevalent repositories were investigated. With the advancement of medical technology, variations of health data are expanding, and new types of health record system can be expected to emerge. The proposed system supports new data variation and new health record in the following ways. First, the system asks IT and healthcare manager or professionals' rating for the latest health record in favour of a few criteria illustrated in Table 5.3. Secondly, the system revises the complete training dataset to relabel the instances. The addition of a new instance does not change the label of the old instances. The class label of the newly added instance is only required to be determined. The system needs only to re-train machine learning algorithms with the updated dataset.

5.5 Conclusion

As more repositories become available for preserving health data, patients will need to select the desired repository. Patients can be expected to avoid choosing a single repository for all their health data because their context of treatment, the pattern of data, legal constraints or personal preferences may change. Therefore, a selection algorithm needs to be developed to automate the storage decision. This is particularly important for continuously streamed health data. In addition, choosing the correct repository is complicated and needs professional knowledge of storage features for interoperability, data security and privacy, infrastructure availability and regulatory issues. Our proposal to disseminate health data among various vendors will prevent the loss of confidentiality and ensure the privacy of medical records if they are stored in one repository. The automated storage recommendation model presented here can allocate health data blocks to a storage medium taking into account data types, data sensitivity, significance and QoP, patient safety and privacy required depending on the profile of an individual.

Chapter 6

The PCA Managed Customized Blockchain Based Framework for Underwater IoT Monitoring

The PCA introduced in this thesis is primarily designed to act as a software agent on behalf of body area sensors and patients for interacting with a Blockchain network. However, Internet of Things spans a range of application areas including eHealth, smart home/city, underwater network, and vehicular network. To explore the applicability of the PCA in common IoT field, we adopted the PCA in managing underwater sensor networks and smart home because Blockchain technologies have not been extensively explored in this field. The underwater network comprises sensors deployed at different levels where sensed data is normally transferred in one direction (towards surface). The functionalities of the PCA have been modified to accommodate Blockchain in underwater sensor networks. Further, the same organization of sensors are applicable for some use cases in eHealth domain. For example, patient monitoring in multi-storied building can require the deployment of sensors at different levels. The aggregation of monitoring data at different layers and rapid transfer of data to a controller such as PCA requires a secure hierarchical routing protocol and the processing data at multiple layers including Edge and Cloud based Blockchain. Since such architectures are more prevalent for underwater sensor networks, this IoT area is chosen to explore the adaptability of the PCA. However, the concepts and changes made in the IoT underwater monitoring framework will fit in the above-mentioned use case of eHealth.

In the previous chapters, we have investigated the Patient-Centric Agent to manage and control data in BC eHealth. The autonomous nature of an agent and higher security and privacy providing BC technology in eHealth applications have motivated us to adopt both technologies in other IoT applications such as underwater IoT, smart home or cities. The Internet of Things has evolved well in autonomous fields such as aquaculture, submarine communication, and underwater monitoring. Also, seafood and fish are in high demand so maintaining the quality and security of aquatic products is important. Emerging technologies like Blockchain (BC) are being undertaken for advancing and monitoring submarine sensor networks. Blockchain technologies can provide new ways of gathering information from underwater IoT sensors and guaranteeing the quality of the products without the requirements for third parties.

In this chapter, we describe a framework of underwater IoT monitoring that adopted smart Gateway Agents to supervise the underwater Internet of Things network. From the architectural point of view, the framework comprises the three layers: Underwater Internet of Things layer,

Edge Network Layer and Cloud Network Layer, each described below.

1. Underwater Internet of Things layer consists of sensors deployed at a different levels of an underwater environment. A level based hierarchical routing protocol using a minimum number of encrypted control packets was devised. The entire monitoring area is divided into sub-areas, each called grid that has a cluster head. All cluster heads in the underwater network accommodate a lightweight Blockchain for managing control key, updating firmware, storing sensor nodes' identifier, and log information. Cluster heads are responsible for forwarding the data packet from the source to the destination based on the level inserted into data packet. The cluster head authorizes its member nodes and maintains the privacy of its member nodes using Bloom filter. A Bloom filter refers to a probabilistic data structure containing a list of pseudonymous identifiers or elements in memory efficiently. The filter is designed to rapidly tell whether an element is present in the list or not.

   The Blockchain on the Underwater Internet of Things does not store data and execute lightweight consensus protocol due to limited power, and memory capacities of IoT nodes. Preferably, all cluster heads hosting BC verify and certify transactions related to log information, security key and node's identity to add these transactions in their local BC ledger. The BC on the cluster heads of Internet of Things layer has twofold merits. The malicious nodes cannot forge the identifier of a registered node. Further, communication flights can be reduced to perform the node's authentication when a sensor node moves to a new cluster head because the cluster head can retrieve necessary information regarding the node from its BC ledger.

2. The Edge layer nodes are at one hop away from the surface nodes of IoUT. The Edge layer houses multiple smart Agents owned by users and different stakeholders to receive underwater sensors data and transmit the data to the Cloud Blockchain. The intelligent Agent on the Edge authorizes the cluster heads in the Underwater Internet of Things layer. Further, the smart Agent affiliated with particular stakeholders governs the mining process on the Cloud Blockchain. In addition, the intelligent Agent selects a group of healthy miners based on their attributes using the TOPSIS method for Proof of Stake consensus mechanism. The significance of using TOPSIS method is that it can generate balanced rating for Miners, considering the weight of each rating attribute. State-of-the-art works linearly combined several attributes to pick Miners for processing transactions and Blocks. However, not all selection attributes, including the reputation, stake, processing , storage and network capabilities of a miner, carry equal significance. For instance, the mining fee of a miner is two or three times important than that of the processing rate for a use case. On the other hand, for different use cases such as firmware update, the reputation of a miner is more important than that of its mining fee. The TOPSIS method allows the system to assign the weight for each attribute and rank the miners by the combined rating of their attributes considering diverse use cases.

3. The Cloud layer hosts the second Blockchain for maintaining the data ledger. The Cloud Blockchain provides the cluster heads in the IoUT layer with the security key and firmware updates via smart Agent placed in the Edge nodes. The Cloud servers perform processing and support permanent storage for IoT data. In this IoUT framework, we described two use cases of IoUT data processing on the Blockchain network. One of the use cases is to detect anomalies from IoT data where a group of Miners selected using TOPSIS method trains a machine learning algorithm and produces transactions with accuracy level. Another group of Miners assesses the outcome and offer rewards to the node producing higher accuracy. The

purpose of this use case is to utilize the Blockchain platform for solving the user's problem without the need of third parties.

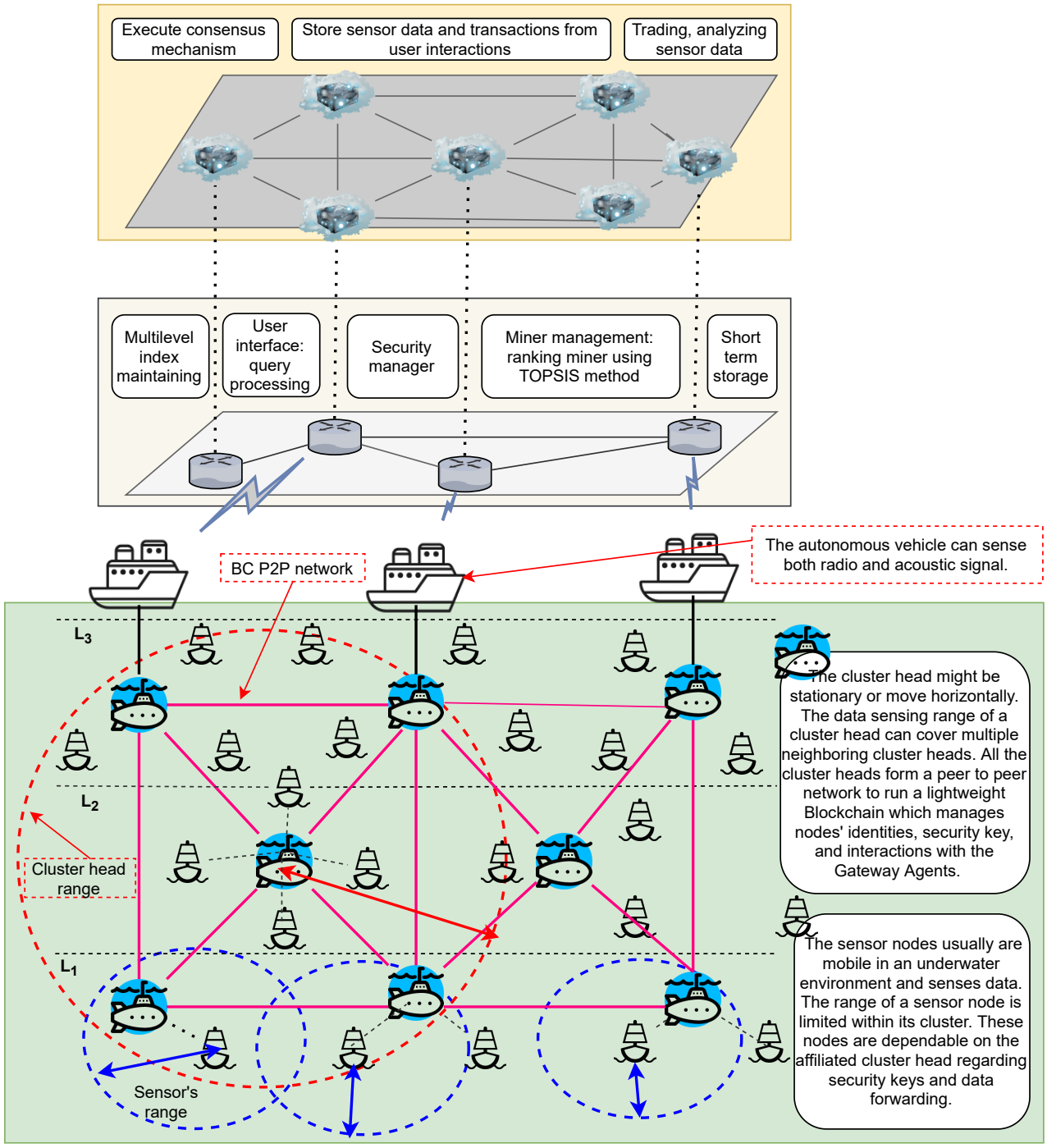


Figure 6.1: The multilayer BC architecture for IoUT monitoring.

In Figure 6.1, a lightweight framework for underwater IoT monitoring is presented. Two different peer-to-peer networks for hosting Blockchain are formed in the Internet of Things layer and Cloud layer respectively. In the Internet of Things layer, only cluster heads accommodate a lightweight Blockchain for managing security keys and routing protocol. In contrast, the Cloud servers maintain a distributed ledger for storing sensor's data and other information including

node's firmware, customers' identities etc. The smart Gateway Agent placed in the Fog network bridges the two different Blockchain networks. We examined the performance of the proposed framework with Proof of Stake consensus algorithm with respect to two use cases. The security and privacy of the framework are also analyzed in terms of known security attacks.

The contents below of this chapter were published in the **Electronics** journal, MDPI in December 2019. The current impact factor of the journal is 2.41. The article has already been cited 15 times (according to Google Scholar)
M. A. Uddin, A. Stranieri, I. Gondal, V. Balasurbramanian,(2019), "A Lightweight Blockchain Based Framework for Underwater IoT", Electronics, MDPI, 8(12), 1552. doi:10.3390/electronics8121552

Abstract

The Internet of Things (IoT) has facilitated services without human intervention for a wide range of applications, including underwater monitoring, where sensors are located at various depths, and data must be transmitted to surface base stations for storage and processing. Ensuring that data transmitted across hierarchical sensor networks are kept secure and private without high computational cost remains a challenge. In this paper, we propose a multilevel sensor monitoring architecture. Our proposal includes a layer-based architecture consisting of Fog and Cloud elements to process and store and process the Internet of Underwater Things (IoUT) data securely with customized Blockchain technology. The secure routing of IoUT data through the hierarchical topology ensures the legitimacy of data sources. A security and performance analysis was performed to show that the architecture can collect data from IoUT devices in the monitoring region efficiently and securely.

6.1 Introduction

Internet of Underwater Things (IoUT) sensor networks [99] enable agencies to monitor underwater spaces to explore resources including gas and gold; measure water temperature; observe fish and oil or gas pipelines; and convey information pertaining to a tsunami, water contamination or other natural disasters [100].

IoUT typically transmit data as sound variations [101]. However, acoustic signals (1500 ms^{-1}) have long propagation delays [101] and high bit error rates. Consequently, underwater communication channels result in low-quality transmission of data. In addition, IoUT devices are deployed in hostile environments and remain unattended, making underwater communication vulnerable to various malicious attacks [102]. Other challenges arising from underwater deployment include dealing with variable water currents, batteries that are difficult to recharge or replace, limited memory, and low bandwidth of devices. These factors in IoUT networks result in an extra barrier to develop a secured routing protocol to collect data from underwater IoT sensors [104].

Various routing architectures and protocols including hierarchical (vertical) [105,106] flat (horizontal) [107], location based [108], multipath routing [109], query based [110], and context aware [111] have been deployed in underwater IoT networks to transmit sensed data to a base station on the surface [112].

The network topology in hierarchical routing is divided into several layers resulting in a compact routing table enabling scalability required for large scale underwater IoT monitoring. Typically, a node with higher energy is nominated as a cluster head (CH) and other nodes with lower energy sense data. The CH's role is to aggregate data and forward the data to the next level. The CH is changed over time due to battery depletion caused by the computational load of aggregating and transmitting data. Further, privacy within a cluster is at risk if the cluster head is

compromised by malicious attacks. Therefore, the cluster head needs a mechanism to protect and maintain member nodes' security and privacy.

A standard protocol in sensor networks known as proactive routing follows a static route but is unsuitable for IoUT devices that change their location with ocean currents [525]. In contrast, reactive routing finds a path on demand by flooding the network with route request messages. Cluster-based reactive routing can be contemplated for large scale mobile underwater sensor networks to address the scalability issue but has high power consumption due to the need to infer the path before forwarding data.

To address IoUT routing issues, we present a lightweight, reactive protocol for hierarchical IoUT networks that require fewer control messages to infer routing the packet forwarding path than other reactive protocols. In this routing mechanism, the cluster head uses a Bloom filter to preserve the privacy of the member node. A Bloom filter contains multiple pseudo-identifiers generated from a node's real identifier using different mathematical hash operations. As a result, the real identity stored in the Bloom filter is hidden from malicious nodes. In a routing protocol, control packets play significant roles in ensuring a node's security and privacy. Nodes from the same manufacturers can utilize symmetric key cryptography to save energy while exchanging control packets. Symmetric key encryption uses the same key to encrypt the plaintext and decrypt the ciphertext. Nodes from different manufacturers need a key exchanging algorithm [526] to communicate between them. The cryptographic hash function that maps data of arbitrary size to a fixed size of code is applied to perform authentication between nodes.

However, the next stage, after collecting IoUT data via a lightweight routing mechanism, is to store and process IoUT data securely. For this purpose, most IoUT architectures are comprised of interconnected underwater IoT devices that transmit data through perception, network, and application layers [118] on the surface, to a remote central, often Cloud-based, server to analyze and store data generated by IoUT devices [111, 119–121]. However, a centralized IoT architecture can be paralyzed by a Denial of Service (DoS) or Ransomware attack because the central server represents a single point of failure and performance bottleneck. To address this issue, a distributed peer-to-peer wireless sensor network has been suggested to store and process IoUT data [121,527]. However, security and privacy are challenged with a distributed peer-to-peer network while processing and sharing IoUT data.

Recently, Blockchain (BC) leveraged distributed, decentralized peer-to-peer networks [107] have emerged to enable underwater IoT data to be stored securely and inexpensively without relying on any intermediary trusted authorities [89] while ensuring privacy. Entities in the Blockchain network participate in processing and validating IoUT data prior to confirming the inclusion of IoUT data to the Blockchain. This process, called a consensus mechanism, substitutes the needs of third-party involvement to process IoUT data. The Consensus mechanism in the Blockchain prevents fraudulent activities and ensure data immutability, transparency, and operational resilience of the Blockchain ledger. Blockchain leverages public key infrastructure (PKI) to authenticate, authorize entities, and encrypt records in peer-to-peer networks. Two entities on a Blockchain network can independently communicate without the aid of intermediaries. The basic data unit of the Blockchain is called a transaction, and several transactions are organized into a Block. Every timestamped Block's header contains a hash code of its immediate, previously confirmed Block. This creates a sequential linkage between data Blocks on the Blockchain, which confirms the irreversibility and ensures that data is tamper-proof.

From the above point of view, Blockchains can facilitate decentralized storage for recording IoUT data and secure processing and sharing IoUT data with different entities. Blockchain technologies have been promoted in IoUT applications that require decentralized access control, stor-

age, and distributed trust [121]. For instance, decentralized Blockchain-based IoUT architecture can be applied for the storage of data generated by many heterogeneous underwater IoT devices deployed at different places, and authenticating software update confirmation for a wide range of marine IoT devices. Further, the consumer or customer often needs to purchase different services such as software update, antivirus, anomaly detection software from a single third party in an IoUT network. In these kinds of applications, the customer needs to trust a single party's solution blindly but the QoS (quality of service) for customers cannot always be guaranteed with a centralized IoUT architecture. The Blockchain technology can be adopted to enhance QoS for these kinds of IoUT applications. However, current Blockchain technology is not computationally efficient for handling IoUT Big data. Multiple nodes in the Blockchain contain a replica of the complete ledger. This distributed feature of the Blockchain demands high storage, and the consensus mechanism, which is the core component of the Blockchain, cannot process transactions faster than a traditional centralized IoUT architecture [121]. The pros and cons of conventional IoUT architecture and Blockchain IoUT architecture are presented in Table 6.1.

Table 6.1: The pros and cons of conventional IoUT architecture and Blockchain IoUT.

Parameter	**Conventional IoUT**	**Blockchain IoUT**
CIA = Confidentiality, Integrity, Availability	Potential risk of unauthorized access to IoUT data. Risk of hidden data alternation or modification by third party [83]. Interruption of service due to outrages [528, 529]. Poor fault tolerance and occurring of bottleneck to handle many service requests.	Preservation of confidentiality because of user-driven record management. Facilitating cross-sharing records ensuring data integrity. Data integrity can be ensured by the replication of the ledger amongst multiple entities replacing third parties' involvement while processing data. Multiple entities can access a record in the Blockchain enabling a system to respond to many service requests
Privacy	Exposing user's identifier to Cloud administrator.	Storage of IoUT record anonymously. Privacy is violated if correlating data to the source [367] is successful.
Freshness	Manipulation of timestamped record by the host. Risk of updating timestamp	Assurance of data freshness using global timestamp.
Cyber Attack	Vulnerable to DoS, ransom, spoofing, and single point of failure due to centralized architecture. Risks of leakage of personal information during data dissemination [424]	Withstand against DoS, ransomware, and single point of failure. Susceptible to long-range attack, nothing at stake, dropping, eclipse, mining attack, and 51% attack.
Cost	High deployment, maintenance, and administrative costs [425]. The integration of fragmented records is expensive	The public can contribute resources. Alleviation of many service costs, including employee wages, legal expenses, and data center rentals.
Interoperability	Poor interoperability due to diverse legal requirements and security methods followed by a different institution. The extra barrier to the cross-border IoUT data sharing [530].	High interoperability because of a universal set of rules and regulations followed by every entity. Support cross-broader IoUT data sharing
Standardization	Standardization already established [427]	Lack of high level standardization [83]

In this article, we merge a customized lightweight Blockchain with an underwater IoT network to provide interested stakeholders with various secure marine services, including secure storage, sharing, and trading platform for IoUT monitoring data. The architecture advanced here implements the Blockchain at the Cloud to accommodate the high computing and storage required for this technology. The framework facilitates aggregating continuous data, indexing, and handling

vast amounts of IoUT data while maintaining privacy and security. We devised a lightweight consensus mechanism that is utilized to validate the processing and analysis of IoUT data. A smart Gateway Agent is envisaged to receive IoUT data from underwater monitoring sensors and insert them into Blockchain.

Our contribution includes the following design elements:

1. The hierarchical IoUT monitoring topology consists of underwater IoT devices at different levels, Gateway Agent in a Fog layer, and Blockchain in the Cloud layer.

2. The secure and lightweight routing protocol transfers the underwater IoT data to a lightweight Blockchain in the Cloud.

3. A lightweight consensus mechanism is proposed to process transactions in the Blockchain. The consensus mechanism selects a group of Miner nodes using the TOPSIS multi-criteria analysis method. Two use cases called IoUT Data Block Inclusion and IoUT Outcome Block Inclusion are investigated using the consensus mechanism.

The related research is described in Section 7.2 before advancing solutions in Section 7.3. The performance of the proposed approach is discussed in Section 6.4 before the concluding remark in Section 7.5.

6.2 Related Work

In recent IoT-based applications, IoT devices collect and forward data to a coordinator node. The coordinator node transmits data to a remote, often Cloud-based server. Some recent Blockchain-based IoT monitoring [366] systems consider a single level of IoT devices. However, some applications such as disaster monitoring, water/air pollution monitoring, and wildlife monitoring require that IoT devices are placed at different levels. In such a topology, an adversary can eavesdrop on the communication and can overhear critical information if secure routing among IoT devices is not ensured. Similar to ground-based IoT, IoUT devices are vulnerable to hole attacks, reconnaissance, Sybil, spoofing, eavesdropping, neighbor discovery, man-in-the-middle, rogue devices, and fragmentation attacks due to large scale, sparse, and unattended networks [102]. In this section, we present a ground-based IoT framework before presenting a comparative analysis of the underwater IoT framework.

6.2.1 Blockchain Based Ground IoT Framework

Some efforts to integrate IoT and Blockchain have been made in recent years. Ali [366] proposed an architecture to incorporate Blockchain and IoT for monitoring smart homes. The IoT data from the smart home is stored, indexed, and shared among consumers using Blockchain. IoT devices have limited processing power, and memory constraints so computationally expensive cryptographic techniques are not appropriate. IoT data can plausibly be generated faster than any consensus protocol can confirm the Block in the Blockchain network [54]. Therefore, Ali replaced the proof of work by a distributed trusted consensus method. However, Ali did not consider securing the routing for the hierarchical topology of IoT applications.

Zyskind [531] proposed a Blockchain-based user protocol for the installation of Apps. In that protocol, the user can set policies and terms, unlike traditional App installation, where the App forces users to agree to some terms and conditions. The same procedures can be applied to use

IoUT data from the Blockchain. Lei [409] suggested a Blockchain-based infrastructure for exchanging keys in a vehicular communication system. The Blockchain's transactions are used to transfer critical information containing a key among heterogeneous infrastructure in a new region whenever the vehicle moves there. Lei's proposal might help to exchange keys in IoT devices in a vehicular network. Tian [532] presented an agricultural food supply chain traceability system by integrating IoT devices such as RFID with Blockchain technology. Tian identified benefits in using RFID and Blockchain technology in a food supply chain to ensure transparency and availability of data generated along the supply chain. Samaniego [533] incorporated Cloud services as a third layer with traditional IoT monitoring to store data streamed from IoT devices.

Recent proposals [534, 535] integrated Fog and Cloud layer to process IoT data. Aazam [534] proposed a smart gateway to integrate IoT with Cloud. The smart gateway can coordinate, preprocess, and manage encryption keys before putting data into the Cloud. Sharma [535] proposed a layer-based architecture for capturing data from IoT devices. The architecture includes IoT devices, a Software Defined Network (SDN) controller in the Fog layer at the edge of the network and a distributed Blockchain in Cloud. The SDN controllers in the Fog layer form a distributed network like Blockchain. The Cloud service providers also constitute a distributed Blockchain. Proof of services that combines the proof of stake and proof of work is proposed as a consensus protocol. We extended the architecture with hierarchical IoT monitoring to efficiently store IoT stream data in distributed Blockchains in the Cloud.

Our architecture differs from these approaches; we designed our architecture for monitoring the underwater environment and presented two use cases of a modified Proof of Stake (PoS) consensus mechanism. We include an energy-efficient cluster-based routing protocol where the node's level is used to forward the data packet to the destination. The cluster head maintains its member nodes' privacy using a Bloom filter. Uddin [527] presented Blockchain-based IoT smart monitoring architecture that includes a network manager to manage keys for the IoT devices. A software agent executing on the Gateway Agent selects a group of Miners by considering their performance to run the Proof of Work. However, routing mechanisms for IoT devices were not discussed in [527].

6.2.2 Cluster Based Routing in Ground IoT

Different routing mechanisms have also been designed for IoT devices [536–538]. Tian [536] proposed an ad-hoc on-demand multipath distance vector routing protocol for IoT. The protocol maintained an internet connecting table (ICT) to discover new neighbors and a routing table for every node. However, the exchange of many control packets is required to discover neighboring nodes which consume much energy and should be avoided for lossy and power-constrained IoT devices.

Chze and Leung [537] advanced a secure, multihop routing protocol where a legitimate service provider inserted encrypted registration information including owner applications, and the network address into the IoT devices before forming a routing mechanism using a "Hello" control message. However, the reliance on a single service provider for registration causes bottleneck problems. The adoption of Blockchain instead of the specific service provider can improve the secure routing mechanism. However, that protocol is not scalable because IoT devices require large storage with a growing number of IoT devices in the network. Iova [538] proposed a destination-oriented directed acyclic graph using some objective functions such as link stability and lifetime. However, the protocol does not support multipath routing, and selections of the neighbor node based on objective functions require the exchanging of a control message.

LEACH [539] and HEED [540] are two fundamental cluster-based routing protocols, and their

variations have recently been proposed in wireless sensor network (WSN). LEACH [539] routing for wireless sensors changes the cluster head after elapsing a designated period of time. LEACH follows a randomized rotation to select a cluster head. The randomization technique creates an opportunity for every node to become a cluster head after a certain period. HEED [540] encompasses the original structure of LEACH utilizing residual energy and node degree or density as a metric for cluster selection to attain power balancing. In HEED, the cluster head is intermittently nominated by two parameters. The probability of a sensor node to become a cluster head is calculated on the first parameter called residual energy, and the second parameter is a function of cluster density, or node degree is used to assess the inter-cluster communication cost. A hierarchical routing mechanism can lead to efficient and scalable IoT monitoring. However, the formation of a cluster head in hierarchical routing consumes a great deal of energy for memory-constrained IoT devices. Therefore, a lightweight cluster formation technique is needed to facilitate scalability.

6.2.3 Underwater Sensor Networks (UWSN)

Multi-hop-based routing protocols that exchange control messages to discover forwarding path [113–117] have been developed for the transmission of underwater data to the surface. However, these protocols drain more power of nodes near the sink.

The depth-based store and forward routing method are considered the most efficient protocol for UWSN. DBR [525] is the first designed depth-based underwater routing protocol that guides packet to the destination solely depending on the water depth of a node. The forwarding node waits for a specified period, which is proportional to its depth to avoid the involvement of multiple nodes to forward the same packet. Later, many depth-based power-efficient protocols [101,104,541,542] have been proposed to address the challenges of the underwater sensor network (UWSN). However, extra battery power is required to estimate the depth of a node in such routing methods. The protocols, as mentioned above, are the subsystems of an underwater monitoring architecture. In this article, we design an end-to-end secure underwater monitoring architecture, including a lightweight level-based routing protocol for UWSN. The Gateway in the Fog layer bridges Blockchain network with the power and memory limited IoUT devices. The Gateway Agent nominates a small set of Miners for executing Proof of Stake consensus mechanism to store and process IoUT data in the Cloud Blockchain. The Cloud can facilitate decentralized and distributed storage, and processing which can tackle privacy and security issues existed in current peer-to-peer distributed underwater wireless sensor networks. A comparative analysis of other existing underwater IoT monitoring frameworks and the proposed IoUT framework is presented in Table 6.2 and 6.3 respectively.

Table 6.2: The comparative analysis of IoUT architecture.

Comparison Criteria	TDSUAC [111]-Secure underwater network adopting security at physical layer, and combining software-defined cognitive and context-aware networks	ALSN [119]-AUV (Automatic Underwater Vehicle) and sensors constitutes networks to monitor and protect underwater pipelines carrying gas or water	HT-SHE [106]-Smart home monitoring with sensors labeled as high, medium and low capabilities	Blockchain IoT-SHM [107]-Blockchain-based architecture for underground structural health monitoring	Proposed Multi-Level IoUT
Fault tolerance	Medium	Low	Low	The centralized Gateway is vulnerable to cyber attacks, thus overall fault tolerance is medium	Multiple Gateway Agents run Blockchain protocols so high fault tolerance capability is achieved
Confidentiality, Integrity and Availability	Preserve confidentiality but cannot guarantee integrity, and high availability	No security standard is defined	Security is defined for the transmission of data but not for data storage, and availability is reduced	Integrity and availability are guaranteed for metadata in the Blockchain	Security, privacy and availability are ensured in both network and Blockchain data management layer
Cyber Attacks	Vulnerable to Ransomware, and DoS attacks	Susceptible to major cyberattacks	Vulnerable to Ransomware, DoS and man in the middle attack	Vulnerable to Ransomware and DoS as data are stored in a single database	Withstand Ransomware and DoS as multiple Cloud servers store the Blockchain ledger
Data Immutability	No	No	No	Single database is used to store data; therefore, there is a risk of data being tampered by an attacker	Yes
Computational Cost	Medium	low	low	PoW consensus mechanism consumes higher power	Lightweight security protocol for routing and PoS, energy and time-efficient consensus mechanism is applied
Secure Generic Communication	Yes	No	Yes	Routing mechanism is not discussed	Multi-level hierarchical routing

Table 6.3: The comparative analysis of IoUT architecture.

Comparison Criteria	TDSUAC [111]-Secure underwater network adopting security at physical layer, and combining software-defined cognitive and context-aware networks	ALSN [119]-AUV (Automatic Underwater Vehicle) and sensors constitutes networks to monitor and protect underwater pipelines carrying gas or water	HT-SHE [106]-Smart home monitoring with sensors labeled as high, medium and low capabilities	Blockchain IoT-SHM [107]-Blockchain-based architecture for underground structural health monitoring	Proposed Multi-Level IoUT
Interoperability	No	No	No	Yes	Yes
Scalability	No	No	Yes	Yes	Routing Protocol is scalable but Blockchain-based storage challenges Big IoUT data
Service Reliability	No	No	No	Smart contracts ensure service reliability	All user's services are verified by the consensus mechanism

6.2.4 Key Management in IoT

The problem of key management for IoT has also been explored in the current literature. Malina [337] examined the performance and memory requirements for IoT devices such as microcontrollers, smart cards, and mobile devices. The symmetric key and hash function are suitable for IoT devices with limited memory and processing power. Jayaraman [543] proposed a modified OpenIoT architecture that used homomorphic encryption techniques. However, homomorphic encryption that requires bilinear pairing, modular exponential, and point multiplication is not appropriate for IoT devices because of the high computational costs involved [337]. Shafagh [373] presented a proximity authentication technique for IoT devices, but proximity authentication is not applicable for mobile and unattended IoUT devices. Lie [383] proposed wireless monitoring and control systems for the smart grid and smart home using a one-time dynamic password for authentication. Ngo [385] described the advantages and disadvantages of maintaining security in wireless networks by using a dynamic key.

We utilized a secure routing mechanism in our designed topology consisting of IoUT devices by maintaining hybrid key management. The secure routing mechanism is not focused on the Blockchain-based hierarchical structure of IoUT monitoring applications. Existing IoUT monitoring systems with a routing mechanism ignore the secure storage of IoUT data. Therefore, there is an opportunity to design a generic hierarchical IoUT monitoring topology with the inclusion of Blockchain for the safe storage of IoUT data.

6.3 Hierarchical Architecture for Underwater IoUT Monitoring

We present an end-to-end framework that handles the collection, processing, and storage of IoUT data. The architecture, as illustrated in Figure 7.2 for monitoring generic IoUT applications, consists of three layers: Internet of Things layer, Fog layer (comprising one or more levels), and Cloud layer. The functional flow diagram of the framework is depicted in Figure 6.3. The top, middle, and bottom parts of Figure 6.3 describe the operation of the Internet of Things, Gateway Agent in the Fog, and Blockchain in the Cloud, respectively. The smart Gateway node is deployed in the Fog layer. The Cloud layer contains a distributed Blockchain that consists of varying Cloud service providers. In this section, we describe the hierarchical routing mechanism that forwards IoUT data to the surface nodes. The IoUT devices are deployed at different levels in the underwater environment.

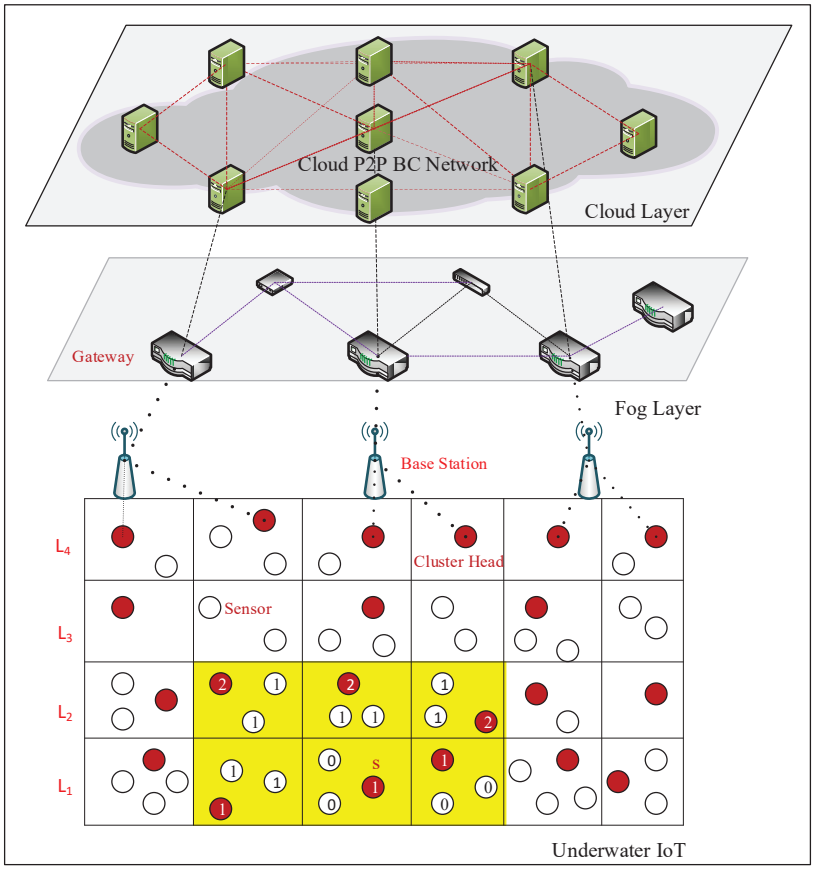


Figure 6.2: The multilevel architecture for IoUT monitoring.

6.3.1 Internet of Things Layer

In this layer, IoUT devices are organized into different levels of the underwater monitoring region, forming a grid network. Each block of the gird contains a cluster head, and members of that cluster are sensors broadcasting data within a specific range. The IoUT device with higher processing power, memory, and stability is normally nominated as the cluster head. All cluster heads form a peer-to-peer Blockchain network. The ledger of this Blockchain stores control and data encryption key, and identities of their member nodes. The following assumptions are made to devise the secure routing in this layer:

Cluster head coverage: A cluster head's range can cover its immediate top, bottom, left, and right rectangular area marked in the yellow shaded cells at the lower end of Figure 7.2. A member node's range is limited to its cluster. The monitoring region is vertically divided into different levels (e.g., the level close to the ground (or base) is 1, the next one is 2, and so on). The relative position of each IoUT device (represented as a dark shaded circle in Figure 7.2) is determined by the level number (L) of its associated cluster head (represented by a red shaded circle in Figure 7.2). The cluster head's level is higher than its member. If the cluster head level is L, the member nodes' level is L – 1 (for instance, if cluster head's level is 4, every member node's level under this cluster head is 3).

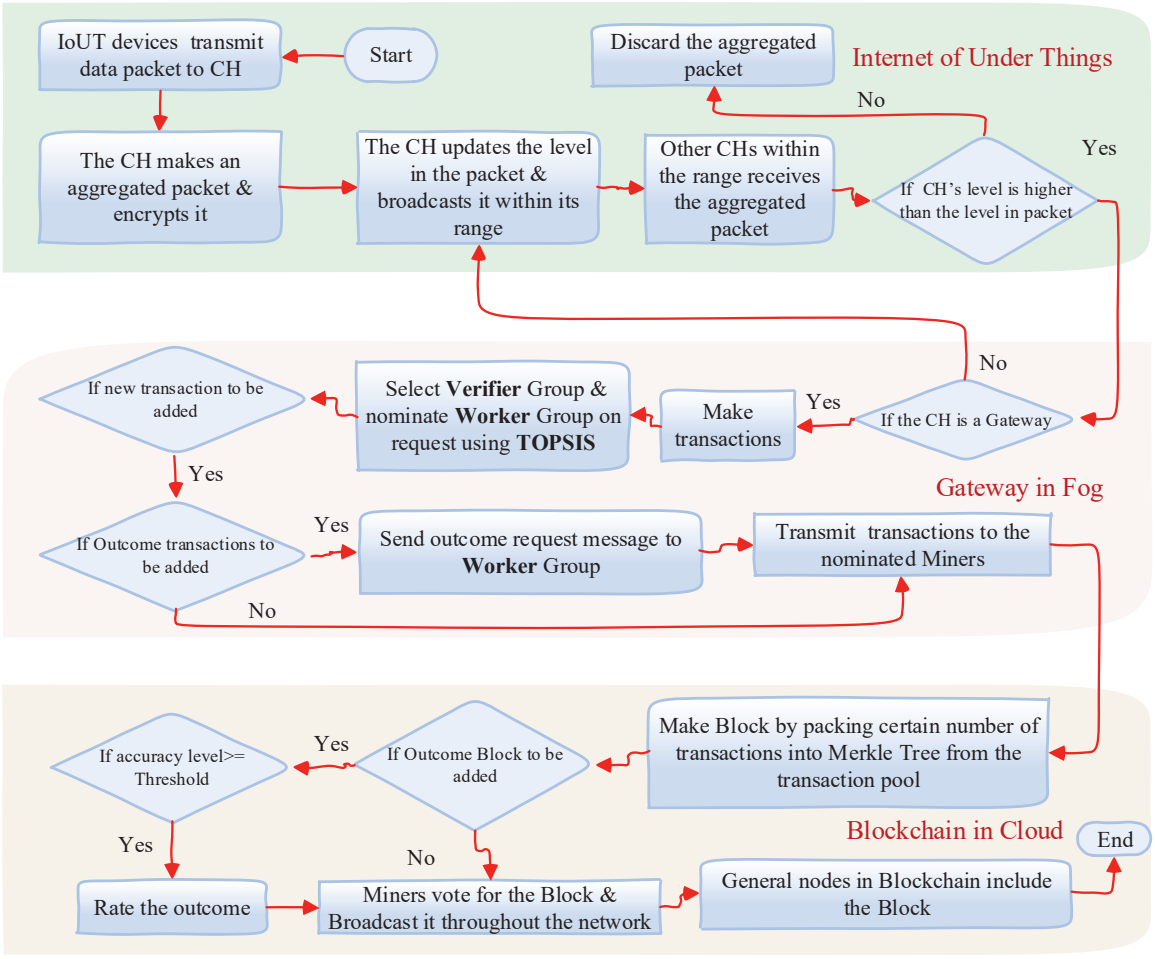


Figure 6.3: The flow diagram for the framework.

Key management: In this protocol, every node uses a common symmetric key, and an asymmetric key (public and private key pairs) for the communication. Every IoUT device obtains the common symmetric key (csk) and the asymmetric key (private key ($privateKey_m$) and public key ($publicKey_m$)) inserted into the node's hardware by their manufacturers. A manufacturer inserts a similar symmetric key and asymmetric key for all its IoUT devices. The public key of each IoUT device is stored on the Cloud Blockchain. The IoUT devices from various manufacturers need to register to the smart Gateway Agent placed in the Fog layer to join in the underwater monitoring. The Gateway Agent verifies the registered IoUT devices by retrieving their public keys from the Blockchain. Later, the smart Gateway Agent regularly obtains the updated keys from the Blockchain and distributes those keys among IoUT nodes of the monitoring region. Here, every node's security key can be periodically updated through Blockchain, according to Lee [544].

Nodes' privacy: Cluster head and member IoUT devices are mobile and can change their levels. Identities of all sensor nodes are stored on the ledger that all the cluster heads maintain in the Internet of Things layer. If a node joins the network under a cluster head, the associated cluster requires to make a transaction containing the new node's identifier. The cluster head broadcast it throughout the Internet of Things layer so that other cluster head can update their local ledger. However, nodes do not utilize their real identifier for performing communications with other nodes. Cluster head and member nodes use a shadow identity (id) with their level to preserve privacy. The cluster head uses a Bloom filter to keep track of its member IoUT devices. A Bloom filter is a

data structure designed to locate or identify the presence of an element in a set rapidly and memory-efficiently. Several hash functions such as MurmurHash3 [545], cityhash [546], or fnvhash can be used to index the hash function in the Bloom filter. For example, the index function for a Bloom filter can be defined as H(id)%m where id is the shadow identifier of an IoUT device, H() is either MurmurHash3 or cityhash function, and m is the size of the Bloom filter.

$$H_1(id) = id\%m$$

$$H_2(id) = (id + 1)\%m$$

A Bloom filter is depicted in Figure 6.4. Suppose the cluster head has two member nodes (identifier 10 and 20 ,respectively) to be recorded in its Bloom filter. The Fnvhash and MurmurHash of node id 10 and 20 are 13 and 14, and 10 and 9, respectively. For node identifier 10, indices 13 and 14 contain 1 (color-marked) and, for node identifier 20, indices 9 and 10 hold 1 (color-marked). The cluster head generates an index from a node id using FnvHash and MurmurHash. If the respective indexes hold 1 in the Bloom filter, the node is a member of that cluster. The benefit of using the Bloom filter is that even If an attacker compromises a cluster head, the identity of the member nodes is not disclosed to the attacker.

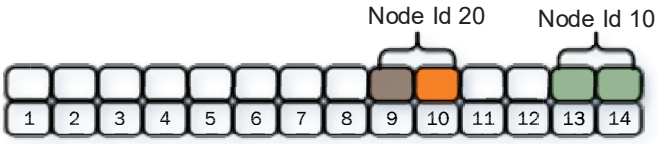


Figure 6.4: The Bloom filter for the cluster head.

6.3.1.1 Data Forwarding Phase

The first step of forwarding data from the source node to destination involves a node's participation in a cluster. A new node needs to send a request control message to a cluster head to join the cluster. The request control message contains the manufacturer's code, residual energy (re), and signature generated by the private key of the node. If the cluster head and the potential member node are produced from the same manufacturer, the cluster head sends an encrypted reply message using the common symmetric key (csk). Otherwise, the cluster head asks the Gateway to provide it with the public key of the member node's manufacturer. The cluster head verifies the member node's signature using this public key. The control message containing a reply holds a data encryption key, identifier(id), and level number (l) for the member node for further communication with the cluster head. This process is illustrated in flow diagram depicted in Figure 6.5.

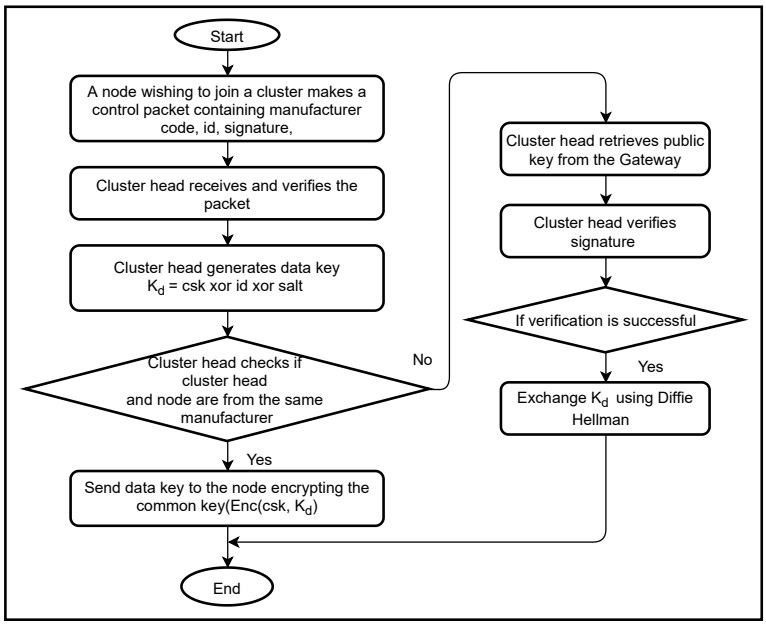


Figure 6.5: The key management process

The cluster head generates data encryption key as follows: $K_d \leftarrow id \oplus csk \oplus salt$ If the cluster head and the member node are from the same manufacturers, K_d is encrypted using the common symmetric key Enc(csk, K_d). This common symmetric key is updated on a regular time basis, and its' ciphertext that is generated by the manufacturer's key is stored on the ledger at Internet of Things layer.

If the cluster head and the member node are from the different manufacturers, they need to use the Diffie-Hellman Key Exchange algorithm to exchange data encryption keys. Similarly, the cluster head gets a data encryption key from the associated smart Gateway Agent for a certain period of communication.

The packet forwarding method is described below. A node transmits its sensed data to the associated cluster header. The format of the data packet is illustrated in Table 6.4. The data packet has two parts, namely the header containing the timestamp (t), node's level number (l), identifier(id), residual energy (re), and HMAC(K_d, t||l||id||re||H(ciphertext)) as a signature; and body containing ciphertext $\leftarrow$ Enc(K_d, data). The cluster head produces HMAC(K_d, t||l||id||H(ciphertext)) to verify the packet's header information such as level (l), identifier, and data contents. If the verification process is successful, the cluster head includes the data into its forwarding packet pool. The cluster head forms a big data packet by taking a certain amount of data from the forwarding packet pool. The cluster head encrypts data using a data encryption key (K_g) given by the Gateway Agent to which the cluster head is associated. The cluster head broadcasts this aggregated data packet throughout its range (marked by yellow rectangular shape in Figure 7.2). The member nodes within the range of this cluster head also receive the aggregated data packet. However, only the cluster heads with a level higher than that of the sending cluster head's level forward the aggregated data packet. Other nodes whose level is equal or lower than the sending node's level discard the packet. The new forwarding cluster head updates the level field by inserting its level before broadcasting the packet further throughout its range. In this forwarding process, only cluster heads (more than one cluster heads at the same level) participate in forwarding a data packet to the destination without the need to exchange any control messages between the sending and forwarding cluster head.

For example, the level of the cluster head at ground level is 1; the level number of its member

nodes is 0. The cluster head of its immediate upper level is 2, and the member nodes' level of this cluster head is 1, which is equal to the level of the immediate lower level cluster head. Now, a cluster head at ground level broadcasts packets throughout the range. The cluster heads that exist at level 2 and to the right and left of the sender cluster head will receive the aggregated packet. The cluster head at level 2 will forward the packet as its level is greater than the sender cluster head (which is at ground level). Other cluster heads and normal nodes discard the packet as their level does not permit sending the data packet. All the cluster heads at level 2 wait for a period that is inversely proportionate to their energy after receiving the aggregated packets ($T = \frac{1}{re} \pm \delta$, δ is random number needed to make each node's waiting time different in case some of them have the same residual energy). The most competent cluster heads at level 2 forward the packets. Other cluster heads overhear the same aggregated packet and discard their copy. The forwarding cluster head will send an ACK control message to the sender cluster head. If the sender cluster head does not receive an ACK within a certain period, the sender will reduce the level of the packet and broadcast the packet again. As a result, other nodes might participate in forwarding the packet. This data gathering method and aggregated data packet forwarding are illustrated in Algorithms 8 and 9. Level-based routing mechanism ensures the involvement of small numbers of IoUT devices in forwarding a data packet without exchanging control messages. However, few control messages are periodically required to exchange security keys among the IoUT devices, the Gateway Agent, and Blockchain nodes.

Algorithm 8: Data Gathering by Cluster Head.

Data: data packet (dp), sender node identifier (id) the level (l) in data packet, cluster head level (L_i)
Result: Data gathering by each cluster head

```
initialize dp = false
while dp is true do
    if L_i > l ∧ HMAC(K_d, t||l||id||re||ciphertext) = Tag then
        if id is in Bloom Filter then
            Store the data packet into forwarding packet pool
        else
            discard the packet
        end
    else
        discard the packet
    end
end
```

Algorithm 9: Data Forwarding Algorithm.

Data: sender cluster head identifier (id), the latest level (l)in data packet, forwarder cluster head node's level L_j

Result: forwarding data packet or reject data packet

```
initialize forwarding = false
while forwarding = true do
    Form aggregated data packet from forwarding packet pool and perform Enc(K_g, data)
    Insert cluster head's level into the packet
    Broadcast the data packet throughout cluster head's range
    for all nodes j = 1 to m within the cluster head i range do
        if L_j > l then
            update the level of the packet
            Estimate T = 1/re ± δ
            if overhear the same packet during T then
                Discard the packet
            else
                Broadcast the packet
            end
        else
            Discard the packet
        end
    end
end
```

Table 6.4: The data packet format.

Sequence No	**Timestamp**	**Level**	**Identifier**	**Residual Energy**
Tag ← HMAC(K_d, t||l||id||re||H(ciphertext))				
Enc(K_d, Data)				

6.3.1.2 Cluster Head Selection

The member nodes under a cluster head include their residual energy every time they send a data packet to the cluster head. The cluster head needs to maintain a database to keep records for its members' residual energy and periodically updates its database. If the current cluster head's residual energy is below the threshold energy or if it leaves the current cluster, the current cluster head selects a member node as a new cluster head considering residual energy and link stability. The current cluster head informs the Gateway Agent about the next cluster head. The current cluster head shares the Bloom filter containing the list of the member nodes with the newly nominated cluster head (the next cluster head), and the newly nominated cluster head increases its level by one after receiving approval from the Gateway Agent.

6.3.2 The Gateway Agent on the Fog Layer

IoUT devices cannot support high processing and memory required for the Blockchain. Further, IoUT devices cannot be directly connected to the Blockchain because the current Blockchain technology cannot process IoUT data in real-time. The Gateway Agent service has been suggested

to bridge IoUT devices and Blockchain. The Gateway Agent needs to be placed at the proximity of IoUT devices to sense, temporarily store, and process IoUT data before permanently adding the data into Blockchain. In this architecture, the smart Gateway Agent is placed at the Fog layer that is close to surface nodes of the underwater network. The reason is IoUT devices require a coordination node that is closer to them to manage the Blockchain on their behalf. The Fog has been called an extension of Cloud to facilitate the computing capabilities to the bottom/edge of the network to provide faster communication, storage, and software services to the lower end devices [547]. Some computing resources, including storage and network services, have already been available on network devices such as a router, switch, and base station that are involved in routing the data produced from end devices. Further computing resources might be augmented to these devices to facilitate computing closer to the user's devices. Fog improves the latency, quality of service, and quality of experience over Cloud because of its proximity to user's devices.

The smart Gateway Agent in the architecture connects IoUT devices with the Blockchain through Base Station, as depicted in Figure 7.2. A Software Agent (SA) executed on the Gateway Agent depicted in Figure 6.6 gathers data from different cluster heads and makes Blocks for the consumers/customers who are interested in purchasing the IoUT data. The Gateway Agent coordinates and manages encryption keys for the Blockchain and IoUT devices. The Gateway Agent in the Fog does not directly take part in selecting a cluster head. Instead, the Gateway Agent provides a cluster head with a data encryption key. The Gateway Agent performs the role of certifying and approving the cluster head through Blockchain. The elected cluster head certifies its members.

Further, the Gateway Agent executes the selective Miner consensus protocol to reduce energy consumption in the Blockchain network. A small set of Miners are randomly selected with Proof of Stake (PoS) to validate the Block. The votes in favor of a Block are collected from the Miners selected using TOPSIS described in Section 6.3.3.3. If the majority of Miners approve the Block, the Block is confirmed in the Blockchain. The small set of Miners equally shares the mining fee for verifying the Block in the Blockchain. In addition, the Gateway Agent also maintains a Bloom filter to record IoUT devices' identities and separates benign and malicious Miners.

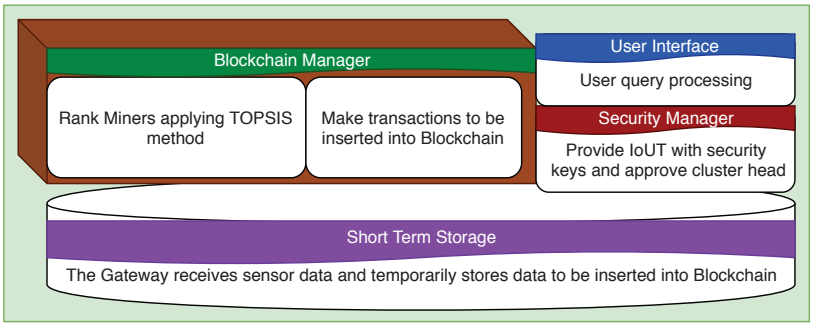


Figure 6.6: The functional block of the Gateway.

6.3.3 Blockchain on the Cloud Layer

In this section, we first discuss the motivation of using Cloud Blockchain to tackle IoUT data. Later, a customized Blockchain to be executed on the Cloud layer is described. The Proof of Stake (PoS) [548] is one of the more efficient consensus mechanisms. With PoS, any Miner can participate in processing Blocks by locking a certain amount of digital coin to the system. We modified this approach by nominating a set of healthy Miners using the TOPSIS method. Further, we present two use cases where the modified PoS can be applied to process IoUT in the Cloud

Blockchain efficiently.

IoUT devices come with various properties such as different manufacturers, security protocols, and power and memory capacities. The integration of heterogeneous IoUT data and the maintenance of updated security protocols for IoUT devices are challenging in the conventional IoUT infrastructure [549]. Further, data generated from different IoUT devices are required to share with multiple stakeholders who are being operated by different legal rules and regulations. The cross border sharing of IoUT data poses the risk of malicious attacks and private information from being compromised. The current centralized architectures have proven not to be so efficient to protect data and disseminate data securely [550] among different stakeholders.

The Blockchain structure can ensure integrity, confidentiality, availability, trust, anonymity, authentication, authorization, user control, non-repudiation, and privacy for the user's record [3]. To be more precise, every transaction in Blockchain contains the user's signature, and Blockchain technology allows others to verify the legitimacy of the user's record. Hence, data provenance is strictly maintained in the Blockchain. Multiple nodes store the complete ledger in a decentralized fashion, which ensures a reliable and tamper-proof storage system. These features of Blockchain technologies have motivated researchers to utilize it to form a secure connection between heterogeneous IoUT devices.

However, the adoption of Blockchain in IoUT needs to address the massive power and memory cost, poor scalability, and legislative compliance issue pertaining to this technology. Blockchain provides IoUT with higher security compromising computation and storage. IoUT devices are power and memory constraint and cannot adopt the Blockchain [23].

The Cloud has been utilized to undertake the high processing and storage requirements for IoUT devices. However, the users need to trust a third-party Cloud service provider that cannot provide them with guaranteed accountability and traceability of their data [551]. This problem can be solved if the Cloud service providers formed a peer-to-peer network and adopt Blockchain technology on that network.

In the framework, we assume that the Cloud service providers support the considerable storage and processing power required for adopting the customized Blockchain. Unlike traditional Cloud services, the customer does not need to trust the Blockchain-based Cloud service provider. Further, public nodes such as computers or smartphones can also participate in validating Blocks.

The components of a standard Blockchain (used in Bitcoin) are described in Figure 6.7 before explaining the customized Blockchain in the next section.

1. The Blockchain operates on a peer-to-peer network depicted in Figure 6.7a consisting of three kinds of nodes: half node, general node, and Miner nodes.

2. The half node and general node produce transactions formatted as in Figure 6.7b and broadcast throughout the peer-to-peer network

3. The Miner nodes collect transactions and pack them into a Merkle tree to form a Block. Each Miner repeatedly inputs the Block into a cryptographic hash function incrementing the nonce field by one until it comes up with a target hash code for the Block. This process depicted in Figure 6.7c is called Proof of Work. Only one Miner that can first publish the Proof of Work for the Block receives rewards.

4. Finally, all nodes except the half-nodes add the Block to the end of the existing ledger, as depicted in Figure 6.7d, by executing the verification process.

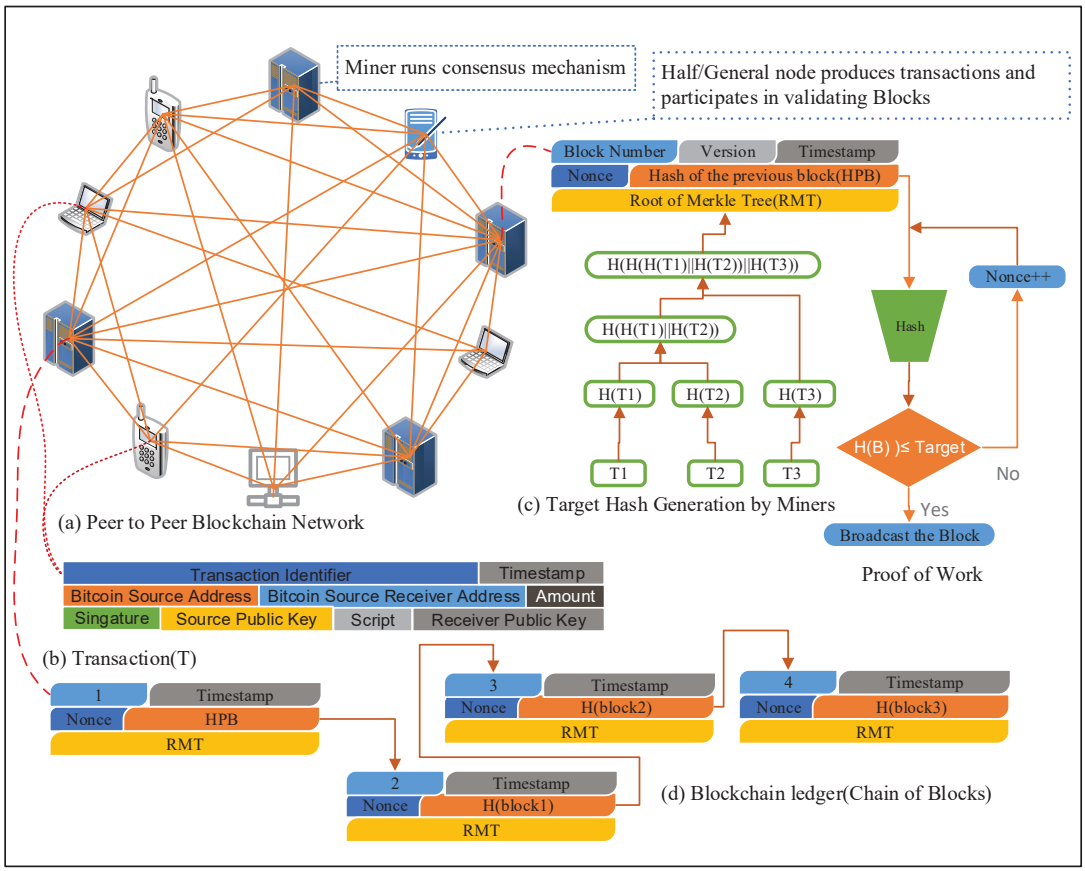


Figure 6.7: The Bitcoin Blockchain.

6.3.3.1 Transaction

A Blockchain transaction is made by the Gateway Agent when it receives a data packet from IoUT devices or other devices. A transaction format is represented in Table 6.5. The Gateway Agent makes different kinds of transactions such as IoUT data transaction, solution or outcome transaction, and financial transaction. A transaction may involve a large volume of data. Thus, the transaction on the Blockchain contains a pointer to the raw data. The sender indicates the Gateway Agent and the consumer indicates persons or organizations who purchase IoUT data. Public/private key is used to generate the signature, and the script describes the procedure for evaluating the transaction (e.g., verification of signature and the legitimacy of sender and receiver).

Table 6.5: The general format of transaction.

Transaction Identifier			
Transaction Type			
IoUT data or Pointer to Data			
Sender Address		Consumer Address	
Signature	Sender pubKey	Script	Consumer pubKey
Block Number			

6.3.3.2 Data Block Structure

The structure of a data Block is presented in Table 6.6. The Block has two parts: header and Data. In the header, Block Type represents the data type on the Block. For instance, the data Block contains records monitored by IoUT devices, and the problem-solution Block holds the transactions of a probable solution or outcome. The outcome-based Block is elucidated below. The Rate Vector field of the Block contains the rates given by Miners based on the accuracy of each solution. All transactions of a Block are organized in a Merkle tree. The Merkle tree ensures the integrity of the data and facilitates Miner to check the integrity without downloading each transaction in the Block.

Table 6.6: The format of data Block.

Block Header of Blockchain	
Field	Description
Version	Block Version Number
Block Types	IoUT data, or Problem solution, outcome, financial blocks
Previous Block Hash	Link for connecting Block
Timestamp	Block Creation time
Merkle Tree Root	Holds the root of Merkle tree to preserve integrity of transactions
Vote	Miner verifies the Block and votes
Rate Vector	Rating for problem solution/outcome transaction
Gateway Agent Signature	Source signature
Index Record	Holds outer and inner index information

6.3.3.3 The Lightweight Consensus Mechanism

In this section, we first discuss the procedure for selecting Miners for the proposed lightweight consensus mechanism. Secondly, we present two use cases where the consensus mechanism can be applied.

The Gateway selects a group of healthy BC (Blockchain) nodes to make the consensus process faster and reliable. This group of nodes takes part in the mining process on the Cloud Blockchain. The Gateway intends to nominate a group of Miners based on multiple criteria of a BC node. The criteria might include reputation, attack vulnerability, amount of locked coin, processing power, storage capacity, availability, throughput, and mining cost; network capabilities such as bandwidth and computing; and other criteria. An efficient method to combine multiple criteria to discover a set of qualified BC nodes to perform mining for IoT data is necessary. In this article, we propose to use TOPSIS [552] (Technique for Order of Preference by Similarity to Ideal Solution) to rank BC nodes. TOPSIS, a multi-criteria decision analysis method, estimates the shortest geometric distance from the ideal best value and the longest geometric distance from the ideal worst value. Before applying TOPSIS, each selection criterion is weighted using AHP (Analytic Hierarchy Process) because each criterion mentioned above should not be considered as equally important. For example, reputation is two times more important than the amount of locked coin or stake; attack vulnerability is three times more desirable than storage capacity or availability; throughput is two times more expected than mining cost; and so on. AHP assigns a weight to each criterion using pairwise comparisons of the attributes or elements. TOPSIS-based Miner selection process is described below.

- Step 1: The BC peer-to-peer network is partitioned into several zones. Primarily, a Gateway randomly picks a certain number of nodes from each zone.

- Step 2: A rank evaluation matrix consisting of m randomly selected nodes from each zone as alternatives and n criteria is created. Each value in the matrix is identified as x_{ij}. The rank evaluation matrix is represented as $(x_{ij})_{m\times n}$.

- Step 3: Each value in the matrix is normalized according to the following formula: $r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{k=1}^{m} x_{kj}^2}}, \ i = 1, 2, \ldots, m, \ j = 1, 2, \ldots, n$

- Step 4: Each value of the matrix is normalized according to the following formula: $t_{ij} = r_{ij} \times w_j, \ i = 1, 2, \ldots, m \ j = 1, 2, \ldots, n$. Here, w_j, which indicates normalized weight for a criterion, is produced using AHP.

- Step 5: The worst alternative (A_w) and the best alternative (A_b) are chosen from each column (criterion) of the matrix. For example, for the criterion reputation, the best ideal value (alternative) is the maximum value of that column and the worst ideal value is the minimum value of that column. For criteria locked coin, processing power, storage capacity, availability, throughput, and network capability, the best ideal (alternative) value and worst ideal value are determined using the same formula. In contrast, for criteria attack vulnerability and mining cost, the best alternative is the minimum value of the respective columns and the worst alternative is the maximum value of the respective columns.

- Step 6: Geometric distance is estimated between the target alternative i and the worst condition A_w and A_b respectively as follows: $d_{iw} = \sqrt{\sum_{j=1}^{n}(t_{ij} - t_{wj})^2}, \ i = 1, 2, \ldots, m$ and $d_{ib} = \sqrt{\sum_{j=1}^{n}(t_{ij} - t_{bj})^2}, \ i = 1, 2, \ldots, m$.

- Step 7: Rank is estimated according to the following formula: $s_{iw} = \frac{d_{iw}}{d_{iw+d_{ib}}}, \ i = 1, 2, \ldots, m$

- Step 8: The Gateway selects a certain number of nodes following the ranking generated by using the TOPSIS. The TOPSIS process is illustrated in Figure 6.8.

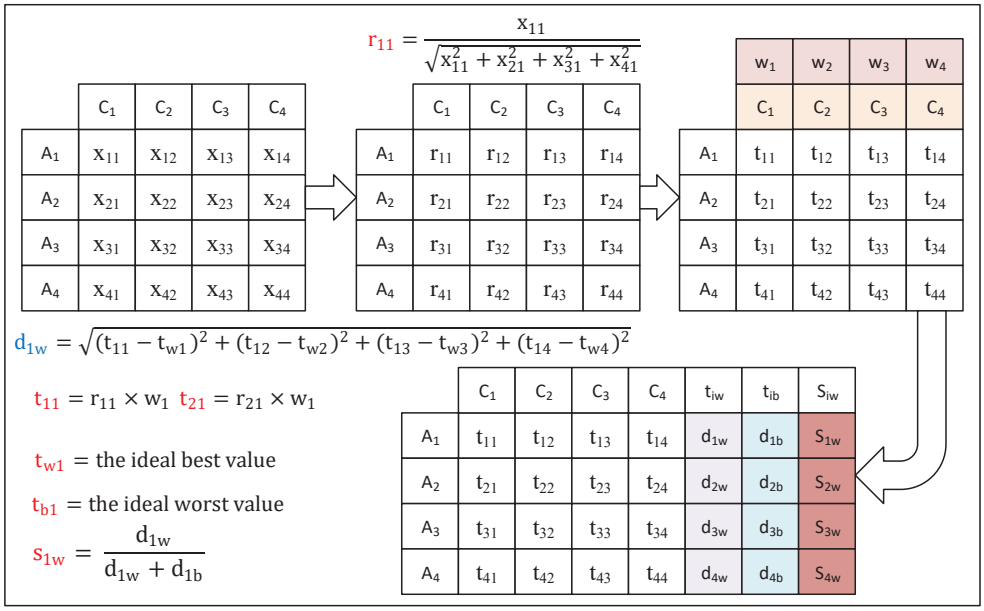


Figure 6.8: The TOPSIS to select a group of Miners.

IoUT Data Block Inclusion: The following consensus mechanism is executed by the Miners to add a new IoUT data Block in the Blockchain. The procedure depicted in Algorithm 10 is described below.

- First, a group of Miners is nominated by the Gateway Agent to verify the newly created Block using the TOPSIS selection method described above.
- Second, Miners mainly verify the hash value of the immediate previous Block, the integrity of all transactions packed in the Merkle Tree of the Block, and the Gateway Agent's signature.
- Third, The Block is broadcast throughout the Blockchain network if specific numbers of the nominated Miners approves the Block as a valid Block.
- Last, the general node adds the Block to the end of the current chain of Blocks.

IoUT Outcome Block Inclusion: The user can request different kinds of services related to IoUT data management from a Gateway Agent. The Gateway Agent solicits services from various entities that participate in managing the Blockchain network. The Gateway utilizes a distributed trust feature of the Blockchain to record the services in the Blockchain. The Gateway Agent initiates the following consensus mechanism when it needs to add a service outcome or a problem solution such as classification or clustering for anomaly detection. The use case depicted in Algorithm 11 is explained below.

- First, the Gateway Agent nominates two small groups of Miners using the TOPSIS method.
- Second, the first group called **workers** generates their respective outputs on a given problem or proposes a solution to solve a problem, and the second group of Miners called **verifiers** rate the results or answer.

- Third, the **verifiers** accept a result of a developed solution only if the outcome from the solution does not exceed a threshold level of accuracy. The **verifiers** rate the solution/outcome based on the accuracy level.
- Last, the **workers** that obtain higher ratings on the solutions of the problem receive rewards for this.

For instance, the Gateway Agent solicits an efficient method from a group of Blockchain nodes (**workers**) to classify IoUT data as malicious and non-malicious data. The solutions from different **workers** will be packed into a Block and transmitted to the verifier group (also nominated by the Gateway Agent). The **verifiers** test the solution using their test dataset. The **verifiers** measure the accuracy level for each solution and rate the solution according to the accuracy level. The **workers** that obtain the highest rate for its solution will be financially rewarded. In such an application, the Gateway Agent does not rely on a third-party or centralized server for confirming a solution. The centralized server system may suffer from higher latency in such an application. In contrast, the Blockchain executing on the distributed peer-to-peer network ensures quick response on the user's task avoiding a single point of failure.

Algorithm 10: Data Block Validation.

```
Data: Block (B)
Result: Confirmed Block, Winner Miner
initialize sigStatus = false, dataStatus = false, prevBlockHash = false, voteCount = 0,
  indexStatus = false;
The Gateway Agent selects a group of Miners(M) using the TOPSIS method;
for each Miner i = 0 to M do
    while newBlock = true do
        sigStatus ← checkSignature (B);
        dataStutus ← checkDataIntegrity (B);
        prevBlockHash ← checkPrevHash (B);
        indexStatus ←checkIndexRecord (B);
    end
    if sigStatus == true ∧ dataStatus == true ∧ prevBlockHash == true ∧ indexStatus == true
      then
        voteCount + +;
    end
end
for each node j = 1 to N do
    if voteCount ≥ thresholdCount then
        Insert the Block into the current ledger;
        Broadcast throughout the network;
    else
        Block is rejected;
    end
end
```

Algorithm 11: Outcome Block Validation Algorithm.

```
Data: Block(B[t]), worker[n], verifier[m]
Result: Ratings of a Block, Winner Miner
The Gateway Agent makes Block with outcome Transactions collected from worker group
The Gateway Agent transmits the Block to the verifier group
for each verifier i = 0 to m do
    for each Transaction t = 1 to n do
        accuracy ← testSolution (B[t]);
        if accuracy ≥ thresholdAccuracy then
            rating[t] ← rating[t]+ rateEstimation(accuracy);
        end
    end
end
```

6.3.3.4 Multilevel Index on the Blockchain

The replication of the complete ledger is a significant reason for having higher security from Blockchain technology. Ledger replication enables an entity to validate IoT data without relying on the third-party. Further, data integrity and availability are guaranteed by replicating the ledger amongst multiple nodes. The distributed decentralized Blockchain storage facilitates multiple active data access point. The other benefit of the distributed ledger is that Blockchain can withstand the single point of failure, Ransomware, and Denial of Service attacks.

In general, transactions related to results or outcome generated from the analysis of raw IoT data, and the transactions related to software update and firmware are occasionally produced. These kinds of transaction can be replicated among multiple entities.

However, nodes that process data streamed from IoUT require a large volume of storage. Although the Cloud server supports massive storage for the Blockchain, volunteers or storage constrained devices might be reluctant to participate in the Blockchain [54].

Validators in IoUT Blockchain do not require the complete chain of Blocks as in digital cryptocurrencies (Bitcoin) to confirm a new Block. Further, the consumer or customer is most interested in the most recent IoUT because recent data are typically the most significant in real-time IoUT monitoring. In such a case, few Cloud servers are allowed to store the complete ledger, and all other validators cache the most recent IoUT data.

Multilevel index records based on time attribute can be maintained in the Blockchain to track the most recent Blocks. A multilevel Blockchain index is depicted in Figure 7.7. In Figure 7.7, the multilevel index comprises the outer index and inner index. Every index record has a search key and two pointers where one pointer holds the hash code of the previous index record of the same level, and another pointer contains the hash code of the next level index record. The link between index records can preserve record integrity and prevent index records from being tampered.

Outer and inner index use year, month, and day, respectively, as a search key. Block's content can be accessed through metadata representing the header's information of a Block. Every entity in the Blockchain stores the complete multilevel index into the main memory and the most recent chain of Blocks in permanent storage rather than store the whole ledger. Several Cloud servers store the entire chain of Blocks. Consequently, limited memory nodes such as smartphones or laptop machines can take part in performing the verification process.

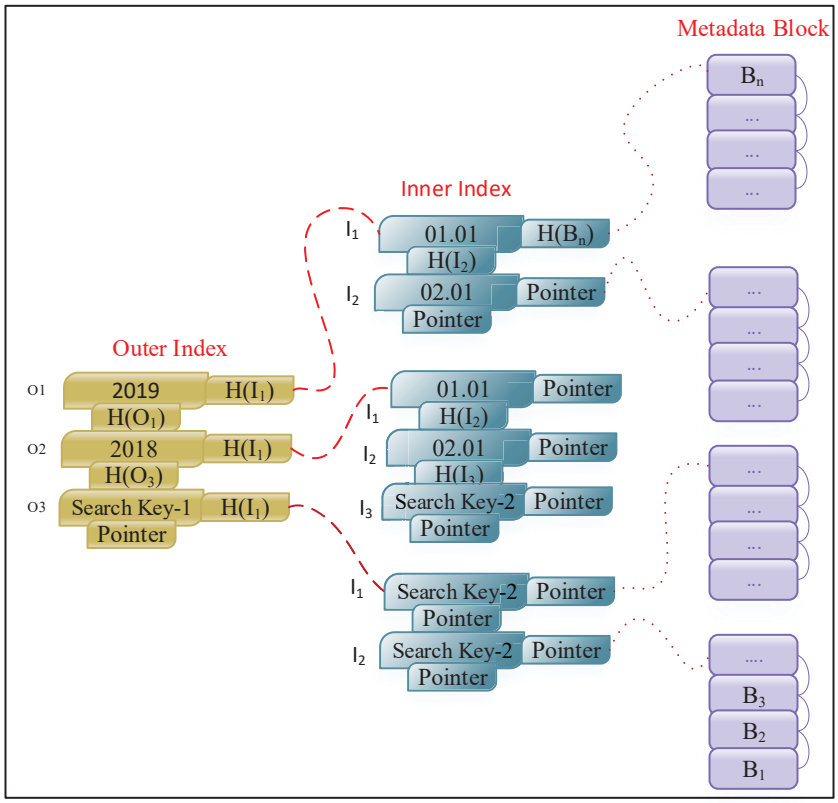


Figure 6.9: Time-based multilevel index record in IoUT Blockchain.

6.4 Performance Analysis

This section contains the performance analysis of the proposed architecture in terms of Block generation energy consumption, time, reliability, remaining energy, and standard security attack.

The architecture was simulated using Java Programming following the iFogSim [128]. The simulation parameter is illustrated in Table 6.7. IoUT devices with different power and memory capacity are considered in the simulation. A private Blockchain module is implemented using the idea introduced in [553]. To implement the second use case of the consensus mechanism that is used to confirm a solution for a particular problem in the Blockchain, we considered a problem of anomaly detection given a dataset. The worker Miners proposed six classifiers, namely C5, C4$\dot{5}$ [554], SVM, Naive Bayes (NB), Multilayer perception (MLP), Random Forest (RF) and other, as a probable solution of the problem. Each worker node proposed a classifier and put the classifier's link in the solution transactions. The Blockchain verifier Miner used KDD Cup 1999 Data [130] to measure the accuracy illustrated in Figure 6.10 of the classifiers by Weka [555]. The threshold accuracy level was assumed as 80%. The Miner accepted the solution, which had a skill above 80%. Here, the worker that proposed C5 with the highest accuracy obtained the highest rating from the verifier and was rewarded.

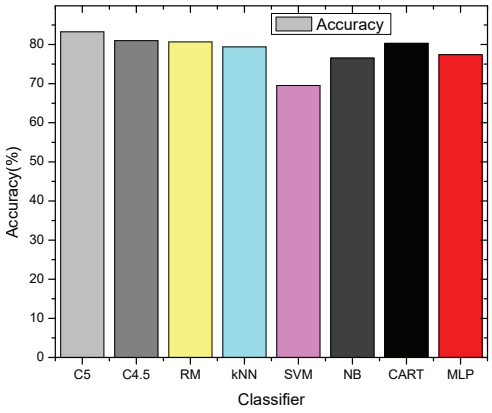


Figure 6.10: The Accuracy of different classifier to detect anomaly.

Table 6.7: The Simulation Parameters.

Simulated Network Area	1000×1000 m^2
Radio Range of a node	100 m
MIPS (Million instructions per second) of Gateway Agent at Fog layer	9900 M–83,000 M
Cloud Node MIPS	317,900 M–113,093 M
IoUT device MIPS	14,000 M
RAM capacity of each node	816 MB
Gateway Agent Bandwidth	600–300 Mbps
IoUT device Bandwidth	100–50 Mbps
Gateway Agent power consumption rate (per Hour)	140–95 W
Gateway Agent power consumption rate (per Hour) while transmitting data	10 W
IoUT device power consumption rate (per Hour)	25–20 W
IoUT device transmission power consumption rate (per Hour)	2
Blockchain Transaction Size	1024 bytes
Block Size of the Blockchain	10×1024 bytes
Instructions required to validate a Block	10 M

The simulation was executed in a 1000×1000 m^2 area. Different numbers of IoUT nodes (100, 200, 300, 400 and 500) were considered for different executions. We analyzed the performances of the architecture concerning the following parameters.

- Energy Consumption: The energy consumption required for the simulated network includes energy for transmitting, receiving transactions, and validation of a certain number of Blocks.
- Block Generation Time: The Block Generation time includes the time required for transmitting, making a certain number (100%) of Blocks, and validating the Blocks.

The simulation was run ten times, and the average value from 10 times simulation was used to represent the performance graph. The energy consumption to generate 100 Blocks using the proposed hierarchical routing under a different number of clusters (10, 15, 20, ..., 50) and nodes is illustrated in Figure 6.11a. The energy consumption increases with increasing cluster numbers as more nodes participate in sensing and forwarding data. However, for a particular cluster number with a variable number of nodes, the energy consumption does not significantly vary. The Block

generation time in BoMLR (Blockchain Oriented Multilevel Routing) is illustrated in Figure 6.11b. Block generation time keeps increasing with higher numbers of nodes and clusters. The fixed number of Gateway Agents experiences some queue latency because of the higher number of associated cluster heads, which reduces the Block per second.

The comparison of Block generation energy consumption and time for the BoML (Blockchain oriented multilevel) and SMH (secure multi-hop) routing are depicted in Figures 6.11c,d, respectively. The SMHR shows higher Block generation energy consumption and time than BoMLR. In SMHR, authentication is performed between IoUT source node and the forwarding node before transferring data. The BoMLR does not need to perform authentication between the source node and the forwarding node before sending data as the node joins a cluster head through authentication. In BoMLR, IoUT devices do not need to generate a new key every time they transmit data to the Gateway. The IoUT devices do not need to exchange the control packet to establish a data forwarding route. Only nodes with higher levels participate in transferring the data packet to the Gateway Agent. Therefore, Block generation delay and energy consumption will be minimized.

Key management and level improve the packet delivery ratio of the proposed protocol. Bitcoin's Blockchain processes around 3–4, and Ethereum [74] processes around 20 transactions per second with Proof of Work that requires a high computational cost. In contrast, Proof of Stake used in our current setting can produce around 2500 Blocks per second. In the proposed architecture, some benign Miners validate the Block in Proof of Stake fashion unless forks are created. Forks on the Blockchain are resolved with Proof of Work and the longest chain rule. The hybrid Proof of Stake will process a higher number of transactions per second than Proof of Work. The PoW consensus mechanism used in the Bitcoin Blockchain demands high computational cost. The Proof of Stake used in the proposed Blockchain does not waste power unless forks occur. Further, Gateway Agent selects a small set of Miners to validate and confirm the Block in the Blockchain. The selection of a small set of Miners also improves the end-to-end energy consumption required for generating Blocks. Specifically, the consensus mechanism of the Blockchain can ensure better QoS for an individual.

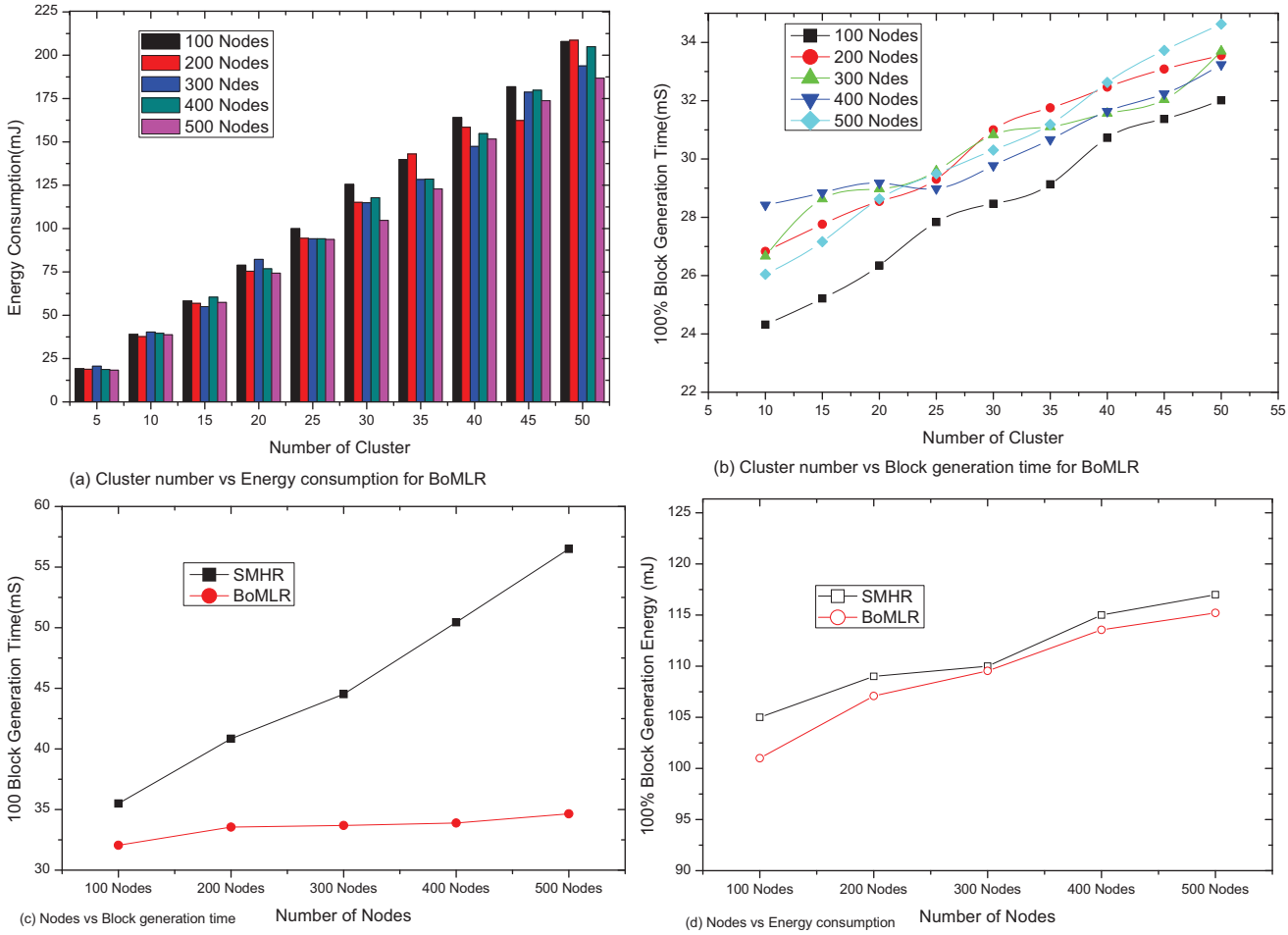


Figure 6.11: The comparison of Block generation energy and time.

The remaining energy comparison of the Internet of Things layer for BoMLR, SMHR [537], HEED [540], and LEACH [539] protocol is illustrated in Figure 6.12a. HEED and LEACH show higher remaining energy in comparison to BoMLR and SMHR because no security is employed in those routing approaches. However, our BoMLR's network lifetime is higher than SMHR. In BoMLR, cluster head selection does not require the exchange of a control message to find the new cluster head as the current cluster head selects the next cluster head. Further, the next cluster head does not need to rediscover its member nodes as the present cluster head shares the Bloom filter holding the member nodes' identifier with the next cluster head. The new cluster head does not need to send a control message containing cluster head information to its members. Member nodes just broadcast their data throughout the cluster, and the cluster head will receive the data as the cluster head's level is greater than its members. Lightweight HMAC verifies the legitimate source node of a data packet. The aggregated data from the general nodes are forwarded to the Gateway Agent by only the cluster heads. The level number associated with each IoUT device guides the node to forward the data packet to the destination. In the proposed framework, avoidance of control message to discover the cluster head and forwarding nodes, involvements of fewer IoUT devices to forward the data packets results in lower power consumption in the IoUT network layer.

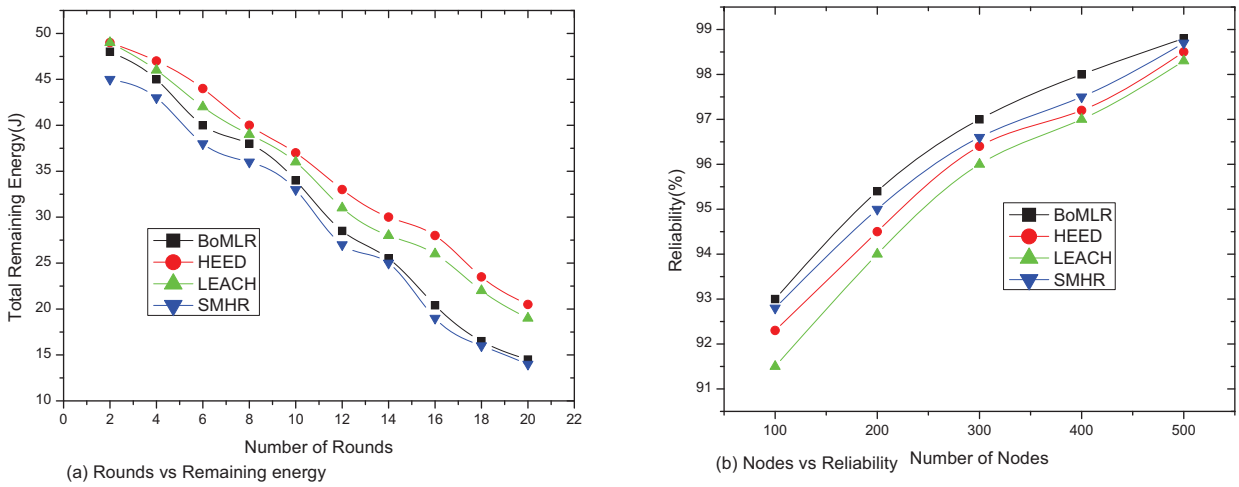


Figure 6.12: The comparison of remaining energy and reliability.

The reliability graph is presented in Figure 6.12b. HEED and LEACH are vulnerable to different attacks because of the absence of a security approach while routing the data packet. SMHR always uses hardware inserted keys. In this approach, if a key is compromised, the packet is always captured by the attacker. On the other hand, in BoMLR, IoUT devices also use hardware inserted symmetric key to transmit data securely. However, IoUT devices obtain firmware updates for the keys from the Gateway Agent through the Blockchain. Thus, IoUT devices are not required to generate new encryption keys regularly. Further, key compromises do not impact heavily on the routing because of updating a firmware through Blockchain.

The IoUT framework may encounter diverse kinds of cyberattacks, and the mitigation of those attacks in light of our IoUT framework are presented in Tables 6.8,6.9 and 6.10 respectively.

Table 6.8: The Security Attack and Mitigation.

Attack Name	**Description**	**Mitigation**
Nothing at Stake attack [556]	Nothing at stake is a situation that occurred in Proof of Stake where a Miner loses nothing misbehaving but stands to be rewarded. In Proof of Stake, a malicious Miner adds a bad transaction in a fork of a Blockchain and all Miners keep validating on two forks in the Blockchain. The malicious Miner waits for the confirmation of the bad transaction and stops validating on the legitimate chain. Thus, the chain holding bad transactions is confirmed and all Miners validating both chains receive reward even one of the chain is discarded.	Proof of Work safeguards Blockchain from nothing at stake attack as a Miner can consume power for only one of the chains at the same time. The Miner can be forced to switch from PoS to PoW when a fork is created in the Blockchain. In our IoUT architecture, a Miner runs Proof of Work when forks occur in the Blockchain until the fork is resolved. Further, the Miner is penalized and enlisted as a malicious Miner by the Gateway if it validates more than one chain.
Long-Range Attack [557]	A long-range attack is a scenario where an adversary creates a branch on the Blockchain starting from the Genesis Block and overtakes the main chain. The reasons for long-range attacks are weak subjectivity, costless simulation of PoS. The weak subjectivity occurs when new nodes or nodes that are off-line for a long time can be fooled to recognize the wrong genesis Blocks or chain. Costless simulation of PoS refers to the ability of a Miner to validate multiple chains of Blocks without spending money or power).	The long-range attack can be mitigated in the following ways: longest rule chain where the Miners overtakes the longest chain, moving checkpoints where only the latest **X** number of Blocks of the chain can be reorganized, and context-aware transactions where every transaction has a field called Block Number that indicates to which Block the transaction belongs. Both Gateway and Blockchain use the above-outlined method to mitigate a long-range attack.
Sybil Attack	An attacker creates multiple false identifiers to control a peer network. In the proposed architecture, the Gateway Agent is vulnerable to Sybil attack because of choosing a group of Miner.	To prevent such an attack, the Gateway Agent might charge high costs from the Miners while creating a new identity and choose Miners based on their trust and reputation. Similarly, mobile IoUT devices require to register to the Gateway Agent to join the peer-to-peer network.

Table 6.9: The Security Attack and Mitigation.

Attack Name	**Description**	**Mitigation**
Eclipse Attack	A hacker controls a large number of IP addresses to make a distributed botnet to overwrite the controlled IP address on the tried Table of the victim nodes and wait until the victim node is restarted. The Gateway Agent is vulnerable to eclipse attack.	The Gateway Agent can verify the IP address registered in the Blockchain and can avoid the impact of eclipse attack
Selfish Mining	Selfish Miners attempt to increase their share by not broadcasting mined Blocks throughout the network for some period and then releases several Blocks at a time, making other Miners lose their Blocks.	The Gateway Agent mitigates such an attempt by selecting a group of Miners from different zones randomly, defining the maximum acceptable time of a Block generation and preparing Blocks with the most recent timestamp.
Replay attack	Replay attack occurred when the attacker intercepts streamed messages to be exchanged between two legitimate entities and delay or resend the stream to one or more of the bodies.	The session timestamp in the data packet prevents an attacker from replaying the same data transaction or control message.
Eavesdropping	Eavesdropping, known as sniffing or snooping attack, is an incursion where an attacker steals information transmitted over a network.	The cluster head sends the encrypted key to its members. The cluster head transmits ciphertext (produced by Gateway Agent's key) of the data to the destination Gateway Agent. The cluster head and the Gateway Agent update their key through Blockchain. All communications occur using a secure channel.
Spoofing Attack	Attackers might change the identity of the source IoUT device, which is called the spoofing attack.	The data transaction contains the hash message authentication code produced from the identifier and other information of the sender. Therefore, a spoofing attack is not possible in the proposed routing mechanism.

Table 6.10: The Security Attack and Mitigation.

Attack Name	**Description**	**Mitigation**
Denial of Service attack	The perpetrator floods or overwhelms a targeted machine by issuing requests and thus make the devices inaccessible to other intended entities. The Gateway Agent might be the target of the DoS attack and results in a single point of failure. The cluster heads associated with the affected Gateway Agent can request other Gateways to resume services for them.	Denial of Service (DoS) attack might occur in IoUT routing. Cluster head and the Gateway may be the target of the DoS attack. Bloom filter helps the cluster head or Gateway Agent to reject packets coming from non-listed nodes. DoS attack does not succeed in the Blockchain because multiple nodes contain the exact copy of the user record and fake Blocks require mining fee to be added into the Blockchain which discourages attacks from making fake Block [54].
Rouge Gateway agent	A rogue agent can compromise a Gateway, or an attacker can pretend to be a Gateway agent. The rogue agent can hold IoUT and decide not to make transactions and send them in the Blockchain network	The Gateway Agent requires to agree on Blockchain protocol suits to act as a broker between IoT devices and the Blockchain network. The consensus mechanism is designed on the standard Proof of Stake in which Miners and the Gateway must deposit digital coin to participate in processing Block in the Blockchain. Thus, a Gateway Agent or Miner discovered as a rouge agent will lose its stake.
Network Analysis	An attack can map a transaction to a particular profile using behavior-based clustering techniques.	Transactions are required to be encrypted so that attack cannot trace and analyze data patterns. Node's identity should be changed to prevent the attackers from tracing transfer history of a token in the Blockchain.
Spreading Illegal Content	Attackers can insert illegal contents into a transaction and broadcast it throughout the network	The smart Gateway provides an IoUT device with a data encryption key. Therefore the Gateway can check the contents of a transaction is before sending it to the Blockchain. However, if the Gateway is compromised and makes encrypted transactions, the Block containing illegal contents is committed in the Blockchain.

6.5 Conclusions

In this paper, we propose a multilayer hierarchical architecture for monitoring and managing IoUT data using Blockchain on the Cloud. The proposed solution organizes sensors into clusters and selects a cluster head based on the residual energy and a node's level to route the data to a higher layer. The cluster head uses a Bloom filter to track member nodes. The underwater IoT routing protocol uses a standard secret key if the sensors are from the same manufacturer. For communicating data with the Gateway, the cluster head uses another secret key, which is provided by the Gateway. Finally, the data are stored in the Cloud using a Blockchain. The smart Gateway in the Fog layer integrates the Blockchain network with the underwater IoT devices to solve the scalability issue, prepare transactions, and route them to Miners. The Block is added to the Blockchain by running a lightweight consensus mechanism. The Blockchain in the Cloud executes the consensus mechanism for inserting IoUT data into the Blockchain. Finally, a performance analysis and mitigation of security attacks analysis demonstrated the feasibility of the architecture to monitor IoUT data.

Chapter 7

The PCA Managed Customized Blockchain Based IoT Framework for IoT Smart Homes

With the increasing spread of IoT, the centralised IoT network structure poses a threat to multiple security vulnerabilities including data forgery, exploitation and unauthorised access to devices by targeting Gateway services[125]. The IoT devices in a smart home are usually connected to the global internet and consumers via Gateway services. Therefore, there needs a security mechanism to safeguard Smart Home Gateway from cyberattacks. IoT devices, and the Gateway normally use the same public/private key pair for signing each transaction on the Blockchain network. This does not provide full anonymity for IoT devices and the Gateway as malicious attackers are able to associate transactions with the source device that uses a similar public key.

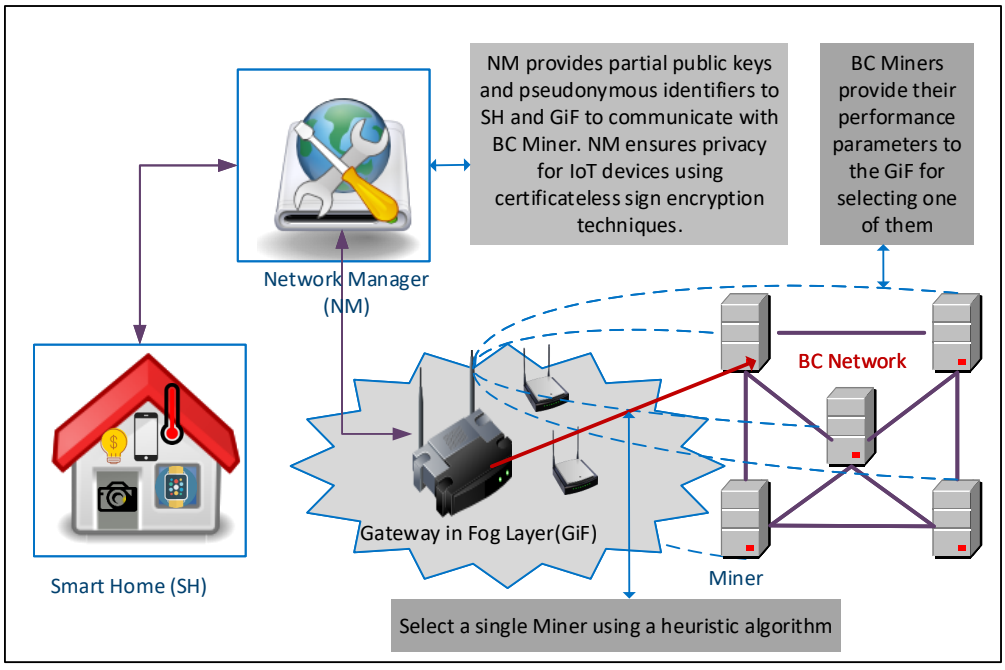


Figure 7.1: The Blockchain leveraged smart home architecture

In this chapter, we described a Blockchain-based smart home monitoring framework depicted in Figure 7.1 which comprises of a smart home equipped with IoT devices, an intelligent Gateway, a Network Manager and Blockchain. The IoT devices encrypted data using certificateless sign encryption techniques and transmit data to the Network Manager that can be owned by the owner of smart homes or central agency. Unlike a third party providing Key Manager, the Network Manager utilizes partial security keys to verify the authenticity of a digital signature of a data packet. The IoT devices and the Gateway request that the Network Manager provides them with a partial public key and anonymous identifiers. Consequently, the Network Manager does not have full control over user's data like other third party agencies. Furthermore, the real identity or data of the IoT devices are not revealed to relay nodes and the Network Manager. The Network Manager forwards the verified packet to the smart Agent on the Gateway deployed in the Edge network and monitors its activities. Likewise, a Miner on the Blockchain plays the roles of Network Manager to forward smart Agent's data to a destination node. The Miner does not know the real identifier of the smart Agent but can validate the Agent's data. This ensures anonymity of IoT devices and the smart Agent on the Blockchain peer to peer network.

A heuristic scheduling algorithm was designed which allows the intelligent Agent to select a group of Miners on the Blockchain and a single Miner from the groups verify a user's Block using certificateless sign encryption approach. A sample Blockchain was developed to analyze the performance of the scheduling algorithm in terms of power consumption and time for generating Blocks.

Since the Gateway in the architecture is centralized, malicious users might be able to compromise the Gateway. A compromised Gateway can provide consumers distorted state of a BC as the consumers access the global state of BC through Gateway. In our smart home architecture, the network manager is responsible for guarding the Gateway and monitoring its activities. In chapter four, we described the decentralization process of Patient Centric Agent at multiple layers. The same strategies can be applied to decentralize the Gateway in this framework.

The contents below of this chapter were published in the 2019 IEEE International Conference on Industrial Technology (ICIT). The article has already been cited 4 times (according to Google Scholar)
M. A. Uddin, A. Stranieri, I. Gondal and V. Balasubramanian, "An Efficient Selective Miner Consensus Protocol in Blockchain Oriented IoT Smart Monitoring," 2019 IEEE International Conference on Industrial Technology (ICIT), Melbourne, Australia, 2019, pp. 1135-1142. doi: 10.1109/ICIT.2019.8754936

Abstract

Blockchains have been widely used in Internet of Things(IoT) applications including smart cities, smart home and smart governance to provide high levels of security and privacy. In this article, we advance a Blockchain based decentralized architecture for the storage of IoT data produced from smart home/cities. The architecture includes a secure communication protocol using a sign-encryption technique between power constrained IoT devices and a Gateway. The sign encryption also preserves privacy. We propose that a Software Agent executing on the Gateway selects a Miner node using performance parameters of Miners. Simulations demonstrate that the recommended Miner selection outperforms Proof of Works selection used in Bitcoin and Random Miner Selection.

7.1 Introduction

IoT devices produce data on a massive scale from many distributed devices. IoT data in smart cities include data from health, transport, productivity, pollution and different community services. Smart cities facilitate real-time monitoring of transport system, health services such as hospital and personal care, environmental management such as noise, air and water quality, strategic planning, better energy management, and improved tourism[416]. In traditional IoT monitoring systems, IoT data is normally transmitted to Cloud based servers for processing. However, traditional Cloud based IoT monitoring is vulnerable to different kinds of cyber attacks including Denial of Service(DoS), and Ransom attacks and represents a single point of failure due to its inherent central architecture. Cloud servers might also be inaccessible due to maintenance or software problems. Further, the closed source code nature of the Cloud creates a lack of trust among vendors and consumers[121]. Blockchain technology enables the collection of IoT data from a large number of devices in order to track, coordinate and store IoT data. Blockchain technology also promotes the creation of many applications such as IoT healthcare that require user controlled access, interoperability while avoiding reliance on a trusted authority[54].

Although Blockchain introduced in digital cryptocurrencies provides an architecture for decentralized storage of IoT data, it requires high computational overhead, long delays, and a great deal of power[287]. This is mainly due to the high computational cost of the consensus protocol to confirm a Block prior to insertion into the Blockchain. Further, the cryptographic techniques and standards to ensure high safety in Block consume a great deal of energy in host devices[54]. Blockchain cannot be implemented over IoT devices due to their power and processing constraints. However, many IoT applications like home automation, transportation, defense and public safety benefit from having a shared repository for the data without relying on a trusted authority to maintain data privacy in Blockchain.

Recent proposals for IoT data collection and monitoring with a Blockchain [54],[535] features a Smart Gateway between the Sensor network and the Blockchain. The Smart Gateway aggregates data transactions into Blocks for storage in the Blockchain. The Gateway might also act as a local Miner which eliminates the requirement of Proof of Works in the Blockchain[287]. However, the elimination of Proof of Work introduces the possibilities of data being tampered by attackers who target the Smart Gateway. Blockchain can enable the data to be stored inexpensively and securely without trusted authorities only if an efficient consensus protocol is ensured[89]. Further, if the Gateway is the entity that always confirms a Block as a Miner in the Blockchain, the Gateway may be vulnerable to a Denial of Service attack. This also introduces a single point of failure.

To safeguard against a Denial of Service attack and a single point of failure, we propose the inclusion of a Network Manager described by [558] between the Sensor Network and the Gateway as a semi-trust center in the proposed architecture. The Network Manager monitors and analyzes the behaviors of the Gateway to safeguard the Gateway from security attacks. The Network Manager also manages encryption/decryption and authentication keys for IoT devices and the Gateway. The Network plays the role of a trusted authority before sending IoT data to the Blockchain. IoT data will be processed in the Blockchain without the involvement of a trusted third party.

According to Uddin et al.[89], not all data generated from IoT devices always requires the highest level of security available. Instead IoT data including medical sensors data might be distributed among different repositories based on the sensitivity, significance and security level required for each stream of data produced from medical sensors according to user's privacy preferences. Uddin et al.[54] introduced an additional role for the Gateway; as a User Centric Agent that determines the storage, security and access level for IoT medical data. They also proposed that a selective Miner consensus protocol can be executed by the User Centric Agent based on the reputation and resources of Miners. However, to design an efficient Miner Selection Algorithm, some performance parameters such as network latency including propagation delay, queue delay, and processing delay, availability and energy consumption of each Miner should be considered.

In this article, we advance an architecture for IoT smart home/cities monitoring. The architecture includes a Gateway to coordinate data flow between IoT devices and a Blockchain. The Gateway also executes an efficient selective Miner consensus protocol to provide the appropriate security of IoT data from smart home or cities in Blockchains. Few studies have addressed the security and privacy challenges while collecting records from IoT devices. In our architecture, IoT devices use Sign encryption to transmit data to the Gateway. The Gateway also transmits the Blocks to the Blockchain Miners using a Sign encryption technique. Sign encryption is a lightweight encryption approach for IoT devices to ensure integrity and confidentiality.

We review related papers in Section 7.2 and describe our proposed architecture in Section 7.3. The performance of the proposed approach is presented in Section 7.4 before concluding the paper.

7.2 Related Works

Ali et al.[287] reported a case study of an application with Blockchain in a smart home. Ali proposed a lightweight Blockchain that eliminates the requirements of executing Proof of Work by introducing a Miner node at the user 's end. The Blockchain based architecture consists of three layers; Cloud storage, overlay and smart home. Proof of Work prevents attackers from tampering with the chain of Blocks. Therefore, the elimination of Proof of Work reduces the security strength of the Blockchain. Biswas[559] proposed a Blockchain based secure framework for collecting information from smart cities. The framework consists of a physical layer that includes

the IoT devices, communication layer that includes communication protocol such as Bluetooth, 6LoWPAN, distributed database layer that is implemented by Blockchain and user interface. The paper did not discuss the basic building blocks of Blockchain and provided no direction regarding the management of huge streams of data from IoT devices in Blockchains. Mengelkamp[560] presented a decentralized private Blockchain based approach for trading and managing the production of renewable energy among local consumers and prosumers. In that proposal, some predefined agents cast their votes on the correctness of the Block as an alternative to Proof of Work. However, this consensus protocol is not appropriate for a public Blockchain without applying some security management or trust center. Sun[561] proposed a conceptual framework for smart cities highlighting the contribution of Blockchain in sharing economic perspective. The conceptual framework includes a service relation between human, technology and organizations. Christidis[387] explored terminology of Blockchain and different consensus protocol of the Blockchain in digital cryptocurrencies. The author focused on the challenges of the combination of IoT and Blockchain. The proposed smart contract[387] which is a set of rules inserted into Blockchain nodes might not be appropriate to be executed in lossy and tiny IoT devices. Stanciu[562] proposed a Blockchain based distributed control system for edge computing. The hyper ledger provided by Cloud services was used as a Blockchain in[562]. The devices in the Edge layer perform computation and processing on data-intensive applications before sending them to the Cloud. The Edge computing reduces the latency and also facilitates storage requirements. Crosby[563] described the basic components of a Blockchain and some finanical applications and non financial applications including notary, and music sectors, and decentralized storage. Neisse[564] discussed data accountability, provenance, scalability and performance of contract based Blockchain applications. Neisse advocated that sensitive data that is not frequently exchanged requires more fine-grained solutions and dynamic data that is more frequently exchanged requires strict scalability and high performance. However, Blockchain's structure to meet the accountability and provenance tracking of data was not discussed in the proposal at length. Ouaddah[565] described the access policies of the resources in Blockchain. Different types of transactions such as grantAccess, getAccess, delegetAccess were used to define the access level of records in the Blockchain. In this article, we advanced a Blockchain based architecture for smart home/cities by ensuring the security and privacy among IoT devices.

Eyal et al. [434] proposed a scalable Blockchain consensus protocol called Bitcoin-NG(Next Generation). In Bitcoin-NG, a leader is elected by using a key block like Bitcoin PoW(Proof of Work) fashion. The leader collects and processes the transactions into blocks called micro blocks by solving a mathematical puzzle(PoW). The consensus protocol reduces the network propagation latency of transactions. However, the process of leader selection consumes energy in Bitcoin-NG. Peterson et al.[435] proposed a random miner selection consensus protocol like MultiChain[436] to elect a miner to perform PoW where miners in the Blockchain take part in the selection process. The nomination of a miner has several advantages including the transmission of transactions to solely the nominated miner obviates the need for distribution of transactions throughout the entire Blockchain network, and the corresponding elimination of wasted computational overhead such as power. However, inefficient miners have a chance to be selected in random miner selection which might increase the latency in the Blockchain. To address this problem, we propose a miner selection algorithm based on a Miner's performance.

7.3 Blockchain based IoT Monitoring Framework

A Blockchain based distributed architecture for smart home/cities/car is shown in Fig. 7.2. The architecture includes smart home/cities/car with IoT devices, Gateway, Blockchain and Network Manager. The Smart home, cities, vehicular IoT and other smart monitoring systems are associated with an individual Gateway and can be connected with a Blockchain through the Gateway.

7.3.1 Internet of Things

IoT devices include mobile, smart watch, temperature indicator, camera and other tiny sensors of a smart home. The IoT devices communicate with the Gateway using Bluetooth or ZigBee protocols. The communication protocol for IoT devices and the Gateway is discussed below.

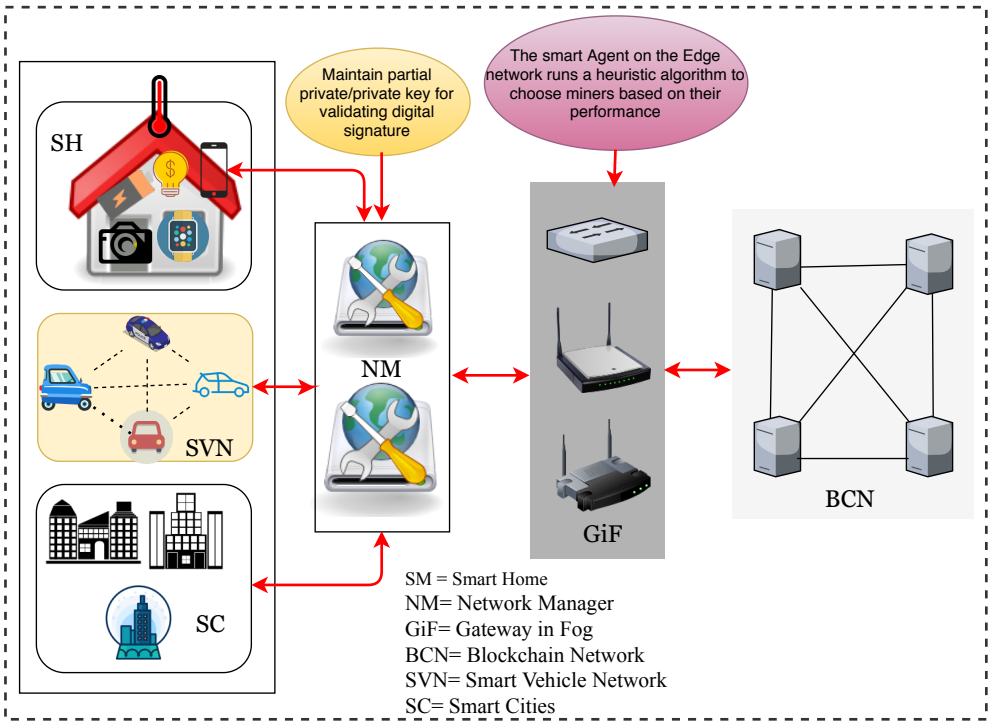


Figure 7.2: The Blockchain based distributed architecture for IoT monitoring

A Secure Communication Protocol between the Gateway and IoT devices is described below. We use certificateless signcryption described by [558] where digital signing and encryption of data are performed by executing a single algorithm. Signcryption is a feasible solution for energy constrained IoT devices to establish a secure communication among them[558]. We describe the protocol for our architecture below.

7.3.1.1 Initialization

The IoT devices and the Gateway initially apply to the Network Manager for registration. The Network Manager provides a partial private and public key to the IoT devices and the Gateway after successful registration. The IoT devices and the Gateway generate their full private key and public key from the partial keys. During registration, the Network Manager(N) provides IoT devices(I), the Gateway(G) and Blockchain node(B) with a pseudonym to enhance privacy.

Next, a Session Key can be exchanged among these entities through a lightweight oneway-hash based exchange protocol proposed in [357]. The session key is updated for future communications using the dynamic key generation as mentioned in [54].

7.3.1.2 The role of the source IoT device

The source IoT device(I) uses CLGSC(ID_S, ID_R, m) to produce encrypted format or signature of message m using a session key exchanged previously between source and destination where ID_S is the identifier of the source(I) and ID_R is the identifier of the receiver. The Certificateless generalized signcryption algorithm (CLGSC), and partial public and private key generation method was described in [558].

1. First of all, If an IoT device with identity I wants to send data(m) to the Gateway with identity G, the IoT device produces message as $M = \mu_I || e_G^I || e_N^G$ where μ_I is the signcryption of data produced by source IoT device and it can be decrypted by the full private key of the Gateway, e_G^I is the encrypted identity of IoT device(I) with full public key of Gateway using certificateless encryption(CLGSC), e_N^G is the encrypted identity of the Gateway with the full public key of the Network Manager(N). Here, μ_I = CLGSC(I, G, m), e_G^I = CLGSC($\varnothing, G, I$) and e_N^G = CLGSC($\varnothing, N, G$). The identity of IoT device and the Gateway are encrypted to enhance their privacy.

2. Next, the source IoT device transfers data and signature generated from the data(δ_I = CLGSC($I, \varnothing, M$)) to a relay node.

7.3.1.3 The role of relay nodes

We presume that some IoT devices might be far away from the Gateway. The IoT devices which are far away from the Gateway transmit data packets using other IoT devices in a multi hop fashion to reduce the higher energy consumption in the IoT network. The data packet(M) from the source IoT device is relayed by other IoT devices as shown in Fig. 7.3. The relay nodes also verify the data signature and insert their signature into the packet. For example, in Fig. 7.3, the relay node(R1) verifies the signature δ_I and produces its signature δ_{R1} = CLGSC($R1, \varnothing, M$). R1 appends its signature with data(M) and relays ($M||\delta_{R1}$) to other nodes.

7.3.1.4 The role of Network Manager

Network Manager is a powerful entity that might be owned by a particular organization such as government institution, or research center that has an interest in monitoring and collecting the IoT data. The Network Manager plays a role in initializing IoT devices of smart home network/smart cities, managing membership of IoT devices, and generating keys. The Network Manager does not need to be fully trusted. The Network Manager handles the problem of key escrow through the generation of partial private key for the IoT devices. In this protocol, when the Network Manager receives the data packet from an IoT device, it verifies the signature and the pseudonym of the Gateway. The Network Manager drops/rejects a data packet if signature verification fails, otherwise the Network Manager directs the data packet to the Gateway. Similarly, the Network Manager filters the data packet destined to IoT devices. Similarly, a Blockchain miner can play the role of Network Manager on the peer to peer network and provides the Gateway with a different partial private and public key as well as a pseudonymous identity.

7.3.1.5 The role of the Gateway

The Gateway receives ($\mu_I||e_G^I$) from the Network Manager, the Gateway first verifies the identity of the IoT device and decrypts the data with its full private key. Gateway also verifies the signature of the IoT device by using their public key. Next, the Gateway processes data into Block($M = \mu_I||e_B^G||e_N^B$) by encrypting Blockchain Miner(B)'s public key and sending μ_I = CLGSC(G, B, b) and (δ_G = CLGSC(G, ∅, M)) to the Blockchain Miner via the Network Manager(N). The Blockchain Miner decrypts the Block and verifies the signature.

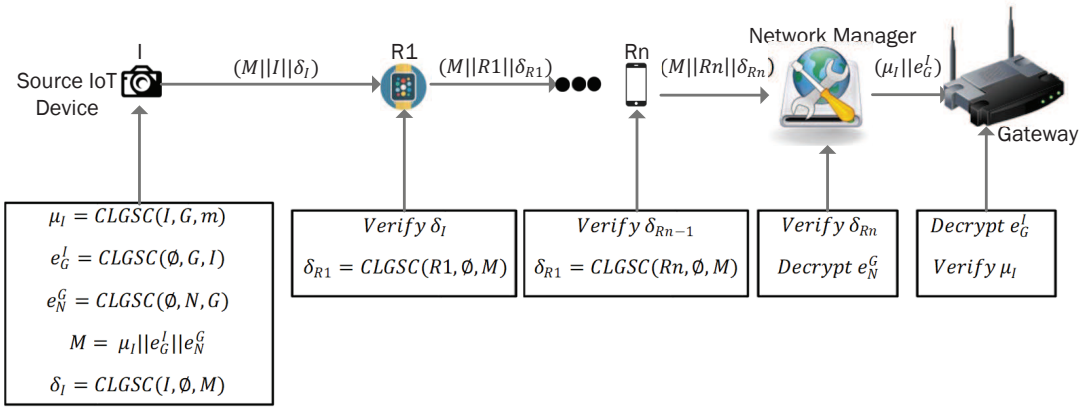


Figure 7.3: The relay process of IoT devices in secure IoT data transmission.

7.3.2 Security Analysis

The advantages of sign encryption are; the IoT devices and the Gateway do not need to fully trust the Network Manager because they receive a partial private key from the Network Manager. The certificateless signencryption facilitates the encryption and generation of a signature to prove the integrity, confidentiality and authenticity of the data using a single algorithm. This reduces the energy consumption of executing two different algorithms for the encryption and signature. The Network Manager reduces the security threat for IoT devices and the Gateway acting as a distributed semi trusted entity. Further, anonymity and contextual privacy of IoT devices(real identity is only known to the intended receiver and eavesdropper is unable to relate data to the source and destination), unlinkability(not possible to link two consecutive transmissions to an IoT device), and forward security which indicates that even if the full private or public key is exposed to attackers, the previous transmissions can not be decrypted because of the use of session keys. The Network Manager might suffer from bottleneck and single point of failure as every traffic to/from IoT devices and the Gateway is directed through the Network Manager. Even if such attacks target Network Manager and impact the normal flow of its transmission, the IoT devices, the Gateway or the intended receiver can request other available Network Manager to provide partial public/private key pairs. The IoT devices can trust any nearby Network Manager as it does not need to generate full public/private keys.

7.3.3 The Gateway

A Fog facilitates the processing of applications on the large number of connected devices at the network edge[566]. Fog computing accommodates computing resources on the network edge devices

such as routers, switches and base station which are closer to the end devices. In this architecture, the smart Gateway that is considered at the Fog Layer gathers some transactions from different IoT devices so that it can support the streaming from real time applications, provide the system with low latency, and location awareness due to its proximity to the IoT devices. The smart Gateway connects IoT devices with a Blockchain. The Gateway coordinates and manages encryption keys for the Blockchain and IoT devices. The Gateway decides which Miner needs to be selected for running the validation process that is needed to add a block in the Blockchain. The Gateway executes a selective Miner consensus protocol to reduce energy consumption in the Blockchain network. The Gateway contains three major modules; Blockchain Management Module, IoT Data Management Module and Security Service Module.

7.3.4 Blockchain Network

Blockchain is a tamper proof decentralized database containing a single truth of user's record. Blockchain reduces the risk of data being modified by attackers because multiple nodes contain the same version of the data[54]. In this architecture, nodes of a Blockchain might be provided by Cloud service providers or the public. The Blockchain's node can be categorized as half nodes, general nodes, benign nodes and Miner nodes. A consumer can access data using Half node such as smartphone. General nodes store blocks and broadcast blocks throughout the Blockchain network for the validation process. The Miners are powerful nodes in terms of CPU processing and memory. The Miner executes the Proof of Work as part of the validation process. The flow diagram of processing a Block in Blockchain is shown in Fig. 7.4 .

1. **Block Preparation:** The Gateway receives data from IoT devices and prepares a Block. The Gateway can decrypt IoT data and put its signature into the Block as it is already registered by Network Manager.

2. **Miner Selection:** A Miner Selection Algorithm is executed by the Gateway. The algorithm nominates a Miner which produces the Target Hash of the Block by consuming its own resources. The Gateway node encrypts the Block's content with destination node's public key and generates a digital signature using the partial private key provided by the Miner so that the Miner node can verify the signature without disclosing identities of the source and destination nodes.

3. **Hash Generation:** The selected Miner inserts the hash of the latest Block of the Blockchain into the **Previous Hash Block** field of the processing Block. The Miner continues incrementing a **counter** which is the only variable field of the Block and inputs the Block into cryptographic hash function until a Target Hash also called Proof of Work is produced. Target hash is a hash code with a certain number of leading zeroes. The Miner broadcasts the Block to the Blockchain network after coming up with the Target Hash. The Miner receives financial incentives for doing this.

4. **Block Verification:** All other nodes in the Blockchain verify the Block to confirm its insertion to the Blockchain.

5. **User Access:** Finally, the consumer retrieves IoT data from the Blockchain for further processing.

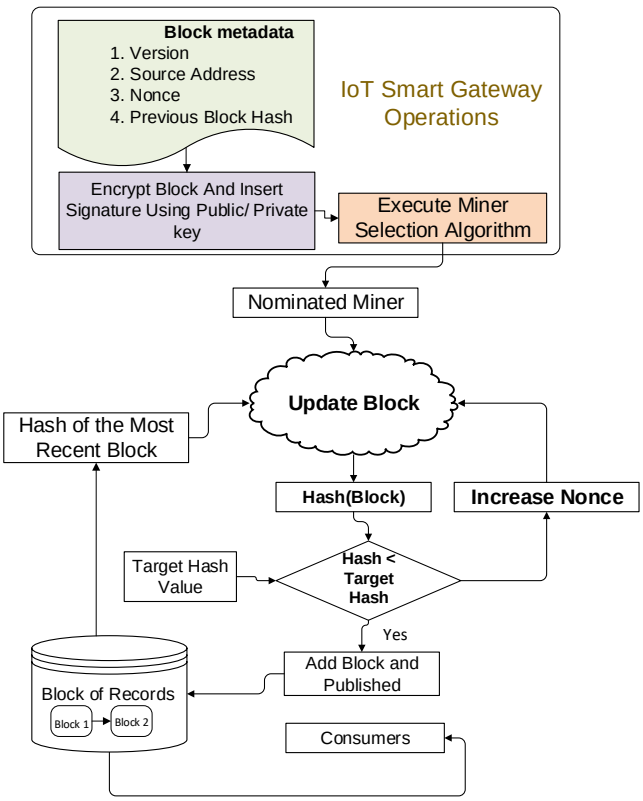


Figure 7.4: The Role of Gateway and Blockchain.

7.3.4.1 Miner Selection Algorithm

In Bitcoin[73], the Proof of Work in digital cryptocurrencies consumes huge processing power because all of the miners compete to be first to generate the target hash of a block to prevent the tampering of records and add the transactions of the block in the Blockchain. We propose to select a group of Miners based on their performance. The miner selection process is illustrated in Fig. 7.5. The prospective Miners provide the Network Manager with their CPU performance, and queue latency in order to take part in mining a block. The Network Manager also assesses the Bandwidth, propagation speed and distance of the communication link between the Gateway and the Miners. The Blockchain Miners communicates with the Network Manager using sign-encryption technique. The network Manager works here as a distributed trust center in the architecture. Further, the Network Manager locks a certain amount of digital currrency of the Miners that take part in the Miner Selection Algorithm so that Miners can not lie to Network Manager about reporting their resources. The Gateway collects some parameters mentioned in[567] including network latency, energy consumption and availability of nodes from the Network Manager. The Gateway aggregates IoT data and builds up a block and executes a selection algorithm presented in Algorithm 12. The algorithm discovers a group of competent Miners. The block is transferred to a miner node listed in the nominated group. The selected miner node runs Proof of Work like Bitcoin[73] and receives its rewards and locked money for doing this. The process reduces the power consumption of Blockchain network as the block is transmitted to only one Miner to produce the Target Hash. The performance parameters estimated by the Network Manager are described below.

Algorithm 12: Miner Selection Algorithm

Data: list of Blocks(n), List of Miners(m), network latency(NM), energy consumption(TE), availability(AV) of all Miners

Result: Scheduling Blocks to the nominated Miners(K)

set used[n] ← 0, set max ← 0
for *each block* i = 1 *to* n **do**
  for *each miner node*j = 1 *to* m **do**
    if *used[j]==0* **then**
      $SM(i,j) = \alpha \times AV_j + (1-\alpha) \times (\frac{1}{NL(i,j)} \times \frac{1}{TE(i,j)})$
      if max < SM(i, j) **then**
        max ← SM(i, j)
        K ← j
      end
    end
  end
  Allocate block(i) to node(K)
  set used[K] ← 1
  if all miners are already selected **then**
    set used[n] ← 0
  end
end

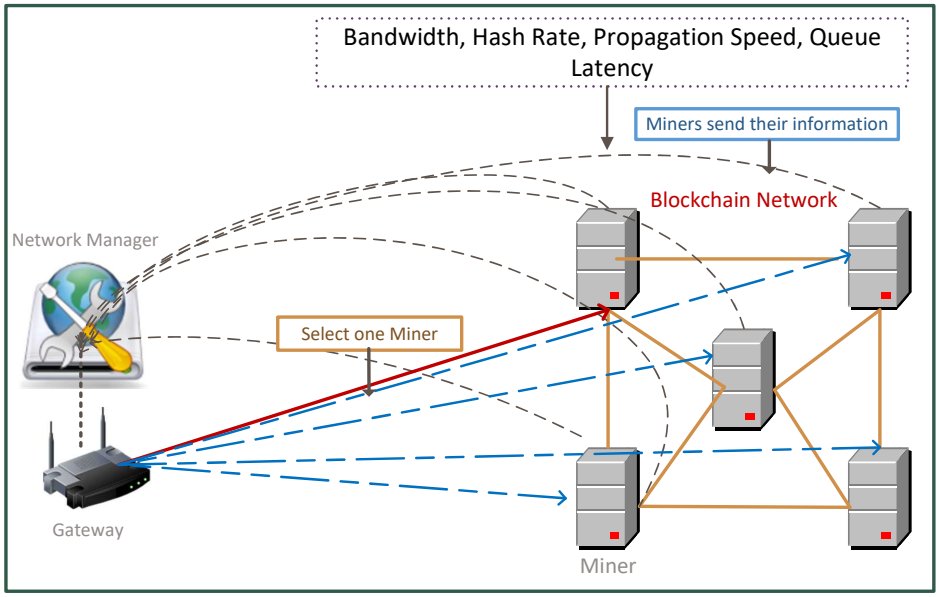


Figure 7.5: The selection of a competent miner.

7.3.4.1.1 Network Latency Network latency is calculated by summing up propagation latency, communication latency, processing latency and queuing latency. The **propagation latency** refers to time required to propagate one bit of the data from the Gateway to a Miner. The propagation latency is crudely proportional to the distance between the Gateway and a Miner. The propagation latency to transfer a block i^{th} from the Gateway to the Miner node j^{th} is computed as follows:

$PL(i,j) = \frac{D_{i,j}}{Prop_s}$ where $D_{i,j}$ represents the distance between the Gateway and Miner node(j^{th}) and $Prop_s$ is the propagation speed of the communication channel between the Gateway and the Miner(j^{th}).

The **communication latency(CL)** is the time that the Gateway requires to get out all bits of the data of a block(i^{th}) on the channel between the Gateway and the Miner node(j^{th}) and it is estimated as follows: $CL(i,j) = \frac{\gamma_i}{B_j}$ where γ_i is the amount of data in the block(i^{th}) and B_j is the available bandwidth of the communication link between the Gateway and the Miner node(j^{th}).

The **processing latency(PrL)** of a block depends on the time that a Miner needs to generate the target hash. The time to generate the target hash of the block(i^{th}) can be estimated as:

$PrL(i,j) = \frac{d \times 2^{32}}{HR_j}$ where d stands for current difficulty level, and HR_j(Hash Rate) represents the number of cryptographic hash operation performed by the Miner node(j^{th}) per second.

The **queue latency(QL)** is a time that a block waits in the queue to be processed. We assume that each miner maintains a single queue to process all the blocks assigned to it. The queue latency is calculated as follows:

$QL(j) = \sum_{k=1}^{T_b^j} PrL(k,j)$ where T_b^j is the total number of blocks waiting to be executed in the Miner(j^{th}) and PrL(k, j) indicates the processing time of a block(k).

Therefore, the **network latency(NL)** for generating hash of i^{th} block in the Miner(j^{th}) can be estimated as in equation (7.1)

$$NL(i,j) = PL(i,j) + CL(i,j) + PrL(i,j) + QL(j) \tag{7.1}$$

7.3.4.1.2 Energy Consumption The Energy consumption of the Gateway includes the energy required to transmit a block to a Miner and its energy consumption during idle time which indicates the time that a Miner node(j^{th}) needs to produce the target hash of the block(i^{th}). So, the required energy for the Gateway to schedule the i^{th} block to the Miner (j^{th}) is measured as follows. $IEG(i,j) = pg_{idle} + (pg_{max} - pg_{idle}) \times T(j)$ where pg_{idle} denotes the rate of the Gateway's power consumption during idle mode and pg_{max} indicates the maximum power consumption rate of the Gateway. T(j) that indicates the response time(Target Hash Generation Time and Queue Latency) from the Miner(j^{th}) is $T(j) = \frac{d \times 2^{32}}{HR_j} + QL(j)$.

Now, the Gateway's energy consumption for transmitting the block(i^{th}) is estimated as follows: $TrEG(i,j) = \rho_t \times \frac{\gamma_i}{B_j}$ where ρ_t denotes the rate of the Gateway's power consumption rate during transmission, B_j is the bandwidth of the communication channel between the Gateway and the Miner(j^{th}).

Now, the Miner's energy required to generate the target hash can be estimated as $ME(i,j) = pm_j \times PrL(i,j)$ where pm_j denotes the power consumption rate of the Miner(j^{th}) to generate the target hash of the block(i^{th}). Therefore, the total energy(TE) for offloading and executing the block(i^{th}) in the system can be estimated as in equation (7.2)

$$TE(i,j) = IEG(i,j) + TrEG(i,j) + ME(i,j); \tag{7.2}$$

7.3.4.1.3 Availability of Blockchain Node(AV) The availability of a node means the amount of time a node is available to process the block. The availability of Miner node(j) is estimated as in equation (7.3)

$$AV_j = \frac{MTBF_j}{MTTR_j + MTBF_j} \tag{7.3}$$

Where $MTBF_j$ and $MTTR_j$ are statistical data, representing the mean time between failure and the mean time to repair respectively for j^th miner node.

In Algorithm 12, we devise a selection metric using (1), (2)and(3) as follows:

$SM(i,j) = \alpha \times AV_j + (1-\alpha) \times (\frac{1}{NL(i,j)} \times \frac{1}{TE(i,j)})$

where higher availability of a miner makes it a better miner, and also the less network latency and power consumption a miner has, the more chance the Miner might be selected for generating target hash. The value of α is a weight factor where $0 < \alpha < 1$.

The Gateway assigns a block to a Miner with a high metric. To avoid the selection of a Miner multiple times, the Gateway prioritizes Miners according to the metric. The Miner with higher priority is selected more than once only if every miner is already selected at least once and there is no available miners to assign the remaining blocks.

7.3.4.2 Data Forwarding from the Gatway to Destination

In Figure 7.6, we suppose that the Gateway (G) wants to send a Block (b) to the destination (D). First, the Gateway nominates a Miner using heuristic scheduling algorithm to write the Block on a distributed ledger and forward it to the D. The other nodes except the selected Miner on BC peer to peer network are labelled as a verifier (V). The Gateway (G) produces ciphertext of the Block (b) using a session key exchanged between source and destination. The ciphertext is μ_G = CLGSC(G, D, b))where CLGSC stands for certificateless generalized sign encryption, G, and D refer to pseudonymous identities of the Gateway and destination node, respectively. The Miner provides the Gateway and destination nodes with partial private/public key and identities. The Gateway encrypt its identity using the destination node's public key (e_D^G = CLGSC(Φ, D, G)) and the destination's identity using the Miner providing public key(e_M^D = CLGSC(Φ, M, D)). The Gateway produces an integrated message ($B = \mu_G || e_G^D || e_M^D$) and a digital signature (δ_G = CLGSC(G, Φ, B)) using its private key to transmit data packet ($B||G||\delta_G$) to BC peer to peer network. The BC nodes except the Miner verify the data packet and tag their digital signature and identities with the forwarding data packet. For instance, the verifier V2 receives ($B||V1||\delta_{V1}$) from verifier V1 and checks δ_{V1}. V2 generates a digital signature using the procedure δ_{V2} = CLGSC(V2, Φ, B) to form a new data packet ($B||V2||\delta_{V2}$). The Miner verifies δ_{V2} and decrypts e_MD to retrieve the destination node. The Miner builds a Block with μ_G and computes target hash code of this Block. The Miner transmits ($\mu_G || e_D^G$) to the destination (D) which can decrypt μ_G. In this scenario, the identities of source and destination nodes are encrypted using the Miner providing partial public/private key. As a result, nodes' identities are not revealed to malicious entities.

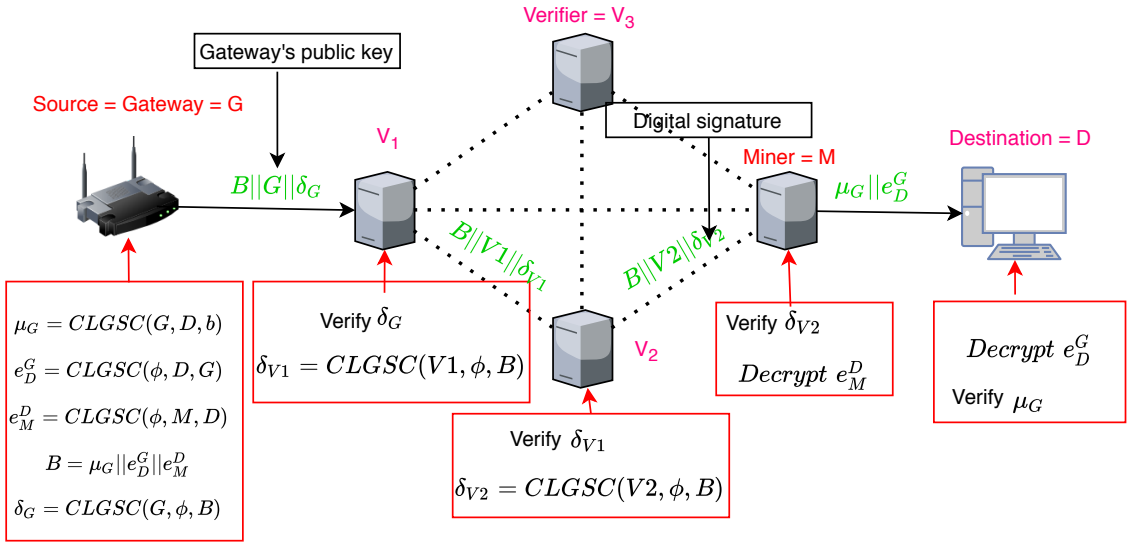


Figure 7.6: The data forwarding from the Gateway to the destination node through Miner.

7.4 Performance Analysis

We implemented a customized Blockchain and Miner Selection Algorithm using Java Programming[553]. We ran our algorithm five times and each time a different number of Miners was considered. We use five machines as the Miners in the simulation. The specification of Miners is shown in Table 7.1. To measure the energy consumption of our customized Blockchain, we use Jolinar [131] which is a Java program to estimate the power consumption of applications at the process level. Later, we normalized the energy consumption of Bitcoin consensus protocol and our selective consensus protocol within the range from 0 to 150 and 0 to 50 joules. Each miner consumes a variable amount of energy according to its specification. The comparison of Proposed Miner Selection(PMS), Random Miner Selection (RMS) and Bitcoin Miner Selection(BMS) is illustrated in Fig. 7.7.

Table 7.1: The Miner Specification

SL No	Memory	Processor
M_1	4.00GB	Intel(R)Core(TM)I3-2310M CPU@2.10 GHz 2.10
M_2	8.00GB	Intel(R)Core(TM)I5-7200U CPU@2.50 GHz 2.71
M_3	16.00GB	Intel(R)Core(TM)I7-4770 CPU@3.40 GHz 3.40
M_4	16.00GB	Intel(R)Core(TM)I3-7100U CPU@2.40 GHz 2.50
M_5	4.00GB	Intel(R)Core(TM)I3-8250U CPU@2.40 GHz 2.50

The difficulty level of generating a target hash is set to 3 for proposed Miner selection as only one Miner is nominated to produce a block. In simulated Bitcoin Blockchain, the difficulty is set to 1, 2, 3,4, and 5 depending on the number of Miners. The reason for setting a different difficulty level in the Bitcoin Blockhain is that the difficulty level is proportionate to the number of Miners in Bitcoin Blockchain.

On the left side of Fig. 7.7, when there is only one miner, our approach showed relatively more energy consumption because the miner selection algorithm consumes some amount of energy. If

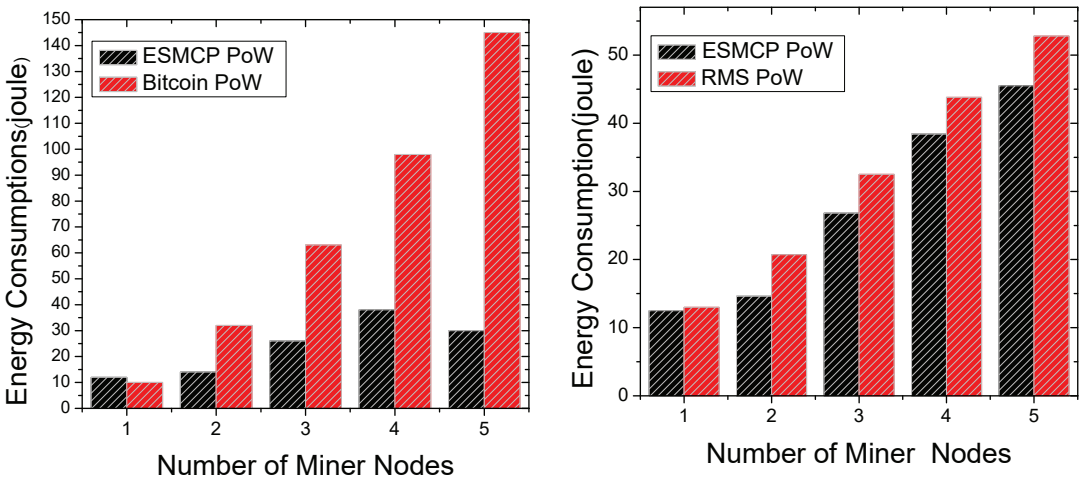


Figure 7.7: The Comparison of proposed Miner selection and Bitcoin Mining energy consumption.

the number of Miners is more than 2, every Miner in Bitcoin Blockchain participates in mining processing. Therefore, energy consumption significantly increases with the number of Miners in the Bitcoin Blockchain. In contrast, the Gateway executes Miner Selection Algorithm based on energy consumption, network latency and availability and nominates only one Miner. As a result, the PMS shows less energy consumption. In the right-side graph of Fig. 7.7, the energy consumption of the PMS and RMS is shown. In RMS, the Gateway selects a Miner randomly. RMS also consumes higher energy than the PMS because in random miner selection, less efficient nodes in terms of power consumption have a chance of being selected.

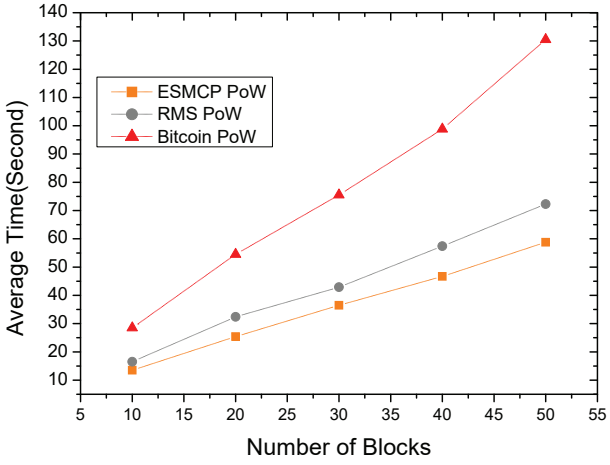


Figure 7.8: The number of Blocks VS Average Time.

The average time required with respect to number of Blocks is shown in Fig. 7.8 for the three miner selection methods(PMS, RMS, and BMS). The proposed miner selection improves much over other two approaches regarding the number of blocks vs. time. The reasons are: the Gateway considers the propagation delay, transmission delay, block processing delay and queue delay to select a Miner and schedules the Blocks in priority basis. The Gateway creates multiple Blockchains for consumers; as a result, some extent of parallelism is achieved. The Gateway can assign its Blocks to more Miners at a time. The Gateway keeps the metadata of the genesis Blocks of every Blockchain associated with a customer if an individual Blockchain is maintained for every

registered user. But in Bitcoin Blockchain, all of the miners compete over the generation of target hash where Miners do not process Block simultaneously.

7.5 Conclusions

Smart monitoring systems need to ensure the appropriate security and privacy while transmitting data to a Blockchain or central server. In the proposed architecture, the sign-encryption technique which is a lightweight cryptography for IoT devices has been used to ensure the privacy and security of IoT devices. We further advanced the functionality of Gateway as a Miner Selector to bridge the gap between power and memory constraint IoT devices and Blockchain. The Gateway selects a small set of efficient Miners to make the Blocks' processing faster. The Network Manager extends the reliability and robustness of the proposed Blockchain based smart cities/home monitoring applications as a semi-trusted center. Miners' selection might introduce a risk that malicious nodes might be nominated to process a Block. Our future work is to design a trust management system to prevent this selection.

Chapter 8

Conclusion and Future Works

8.1 Conclusion

Blockchain technologies ensure higher security in IoT applications without the supervision of third parties. However, the adoption of this technology in IoT applications compromises some degrees of user's privacy, experiences relatively lower throughput, and consumes higher energy than the traditional centralized IoT infrastructure.

In this thesis, we attempted to address these issues while undertaking Blockchain technology in several IoT applications, mainly in eHealth. Our research throughout the thesis found that introduction of a Patient-Centric Agent can effectively meet challenges of Blockchain's integration in IoT eHealth. The Patient-Centric Agent (PCA) can preserve a patient's privacy by implementing health data storage policies and the user's preferences. The PCA decides the storage medium of health data based on diverse contexts which can handle on-chain storage issues of Blockchain.

Meanwhile, instances of the Patient-Centric Agent are deployed at three layers: Sensing, NEAR processing, and FAR processing layer to increase throughput and fault tolerance capacities of the eHealth architecture. The Patient Agent at the NEAR processing layer participates in running a Fuzzy Inference Process-based Proof of Stake consensus mechanism and executing a Blockchain leveraged task offloading algorithm. The operation of the Patient Centric Agent at the three levels demonstrated a significantly improved throughput and security strength over the centralized eHealth architecture. Additionally, the Patient-Centric Agent has the following security and privacy-related functionalities:

1. To implements role-based access control.
2. To generate a dynamic key for voiced based authentication between Patient Agent and medical sensors.
3. To distribute data encryption key among homogeneous Patient-Centric Agent located at different levels using Shamir's Secret Sharing (SSS) method.
4. To ensure the anonymity of the patient using Ring signature and multi-digital signature.

The Patient-Centric Agent guarantees the Quality of Service for patients by selecting appropriate miners for processing patient's health data and storing health data into multiple Blockchains. Furthermore, the proposed Agent was investigated in managing and monitoring underwater IoT data and smart home. The intelligent Agent in underwater IoT architecture maintains two Blockchain

peer to peer networks: a lightweight Blockchain hosted by cluster heads in the marine IoT layer, and a data storage Blockchain in the Cloud layers. The Blockchain in the IoT layer manages the security key for IoT devices and authorizes IoT devices deployed at different levels. A lightweight level based routing protocol was developed to send IoT data to the smart Agent for the underwater IoT layer. The Agent also governs the mining process of the Cloud Blockchain and supports time-based multilevel indexing to allow the lightweight nodes to store only recent IoT data. However, the Agent is vulnerable to different forms of cyberattacks. To tackle this issue, a Network Manager was included in the smart home management. The inclusion of a Network Manager module enables activities of the intelligent Agent to be monitored to safeguard it from malicious attacks. The Network Manager provides the Agent and IoT devices with partial public/ private key to support certificate-less sign encryption technique which is known as privacy-preserving encryption methods. The performance analysis of the proposed Blockchain-based IoT frameworks was analyzed on a private Blockchain using Java Programming and following iFogSim, Java Cryptography library in terms of throughput, Blockchain generation time, power consumption, reliability, communication overhead etc.

8.2 Limitations and Future Works

The works presented in this thesis have some shortcomings that include several unresolved Blockchain and eHealth issues to be addressed in future.

1. IoT devices need to produce ciphertext of data to store that on-chain in the Blockchain. The conventional asymmetric or symmetric encryption techniques involve higher costs and delay to decrypt. There needs a lightweight encryption technique which will be particularly appropriate for IoT devices. Additionally, a new consensus protocol which can validate ciphertext of transactions is required to be designed. The first work in this thesis utilised symmetric key encryption for storing data on-chain. The process is feasible as only one trusted Miner is nominated to process a Block. However, symmetric key encryption technique threatens user's privacy in a distributed consensus mechanism.

   Information processing on the Blockchain nodes encounters the risk of leakage of data as plaintext data is shared and accessed with many nodes. In the BC computing model, the implementation of homomorphic encryption technology has promising potential to secure user data and can allow mining to preserve user's privacy[568]. Homomorphic encryption method allows any third-party service providers such as Cloud servers to conduct certain forms of operations on the ciphertext without first decrypting encrypted data while preserving data privacy at the same time. The integration of homomorphic encryption with Blockchain-based eHealth can potentially protect a patient's privacy in a decentralized model[568]. The future task is to formulate a consensus method that will be consistent with the technique of homomorphic encryption.

2. In the second work, homogeneous Patient-Centric Agents deployed at a different level were sought to process patient data, and thus patient's privacy is preserved. However, other Patient Agents also need to participate in managing health data to ensure the quality of services for the patient. In Blockchain technology, multiple heterogeneous Patient Agents in different service providers can store patient's data. To preserve patient's security and privacy, the source Patient Centric Agent transmits ciphertext (produced using owner agent's public key)

to other foreign Agents. As a result, healthcare providers are unable to access data from their nearby foreign Agents (heterogeneous agents) and cannot benefit rapid access to data.

To solve this issue, our future work aims to adopt proxy re-encryption system. Proxy re-encryption system[173] is a cryptosystem that allows third parties to re-encrypt a ciphertext (This ciphertext was already generated by the data owner using her or her private key) so that an authorised consumer can decrypt it.

In general, a data owner encrypts his data using his or her public key prior to storing the data in a conventional Cloud server. As a result, the owner has to download data from the Cloud storage for sending the data to a consumer. The data owner needs to decrypt the data and encrypt it again using the consumer's public key who wants to access the data. This consumes higher network bandwidth, energy, and experience greater latency as the owner needs to download the data from the Cloud server, every time a consumer requests access to data. To deal with this issue, the proxy re-encryption enables the owner to avoid downloading encrypted data from the Cloud storage. Instead, the owner generates a proxy key using the consumer information and provides it to the Cloud storage that re-encrypts the ciphertext (the owner's encrypted data that was already stored in the Cloud server). The consumer downloads this re-encrypted data from the Cloud server and can decrypt with his or her private key. For example, Bob wants to access Alice's data stored in a Cloud server. Alice generates a "re-encryption" key using Bob's public key. Alice sends this re-encryption key to the Cloud server which re-encrypt Alice's ciphertext using this new key. Bob can finally retrieve double encrypted data, specifically produced for him from the Cloud server. Similarly, a Patient Centric Agent can utilize the proxy re-encryption technique while storing its data in foreign Agents.

3. The current structure of most Blockchain technologies does not allow the storage of health data on-chain due to high requirements of memory. The researchers sought different off-chain storage mechanism to accommodate IoT data and suggested saving the hash value of the data on-chain. Although this approach can solve the storage issue, it cannot guarantee the tamper-proof of the data. To address this issue, the method that was devised in this thesis is to distribute health data among different health repositories, including Blockchain EHR, healthcare professional managing EMR, government providing EHR and proprietary Cloud healthcare considering different kinds of data contexts. However, still, this approach might include a massive quantity of data on the Blockchain. Therefore, a better approach will be explored to address Blockchian storage issues in future.

   Furthermore, the model needs to collect preferences from diverse kinds of users, including individuals, patients, IT experts, and healthcare professionals, which threatens an individual's privacy. Future work is planned to apply Federated Learning[569] in the model, which is a privacy-preserving machine learning approach. Federated learning is known as collaborative machine learning where client nodes train a generic machine learning algorithm downloaded from a centralized server. The clients use their local data to train the algorithm without exchanging their local data with the server. The clients upload their respective modified weight (e.g. weights of a Neural Network) to the server, which estimates the averages of weight sent by different clients to set ultimate weight for its machine learning algorithm. In this technique, clients can preserve data privacy because they do not need to share their local data with a third party providing server. This method contrasts with conventional central machine learning strategies where all local datasets are submitted to a single server and

with classic decentralised methods, which are often assumed to be the same distribution of local data samples. Federated learning allows multiple actors to create a standard, scalable machine learning model without sharing data. Thus, this process may address critical issues such as data privacy, data protection, data access rights, and access to heterogeneous data.

4. The efficiency of the Patient Agent was investigated with respect to a customized Blockchain. Future work is planned to integrate the Patient Centric Agent with Ethereum and other lightweight enterprise Blockchain to evaluate the efficiency and performance of multiple Blockchains in managing patient data. One of the major reasons for Blockchain's poor performance is that every node on the network is involved in the processing of each transaction.

   Sharding[570] was introduced to improve a Blockchain's performance. Sharding is a splitting strategy that distributes computing and storage workloads across a P2P network such that each node is not responsible for managing the entire network's transactions load, but instead handles information related to its partition or shard. In this technique, several Blockchains called a chain of shard are managed by network nodes instead of maintaining a single Blockchain for all transactions. Each shard consists of its own nodes or validators that apply a PoW or staking or voting consensus mechanism. This sharding approach has been partly adapted in the consensus mechanism of our proposed decentralized eHealth architecture. The proposed consensus mechanism partitioned Fog peer to peer network into different cluster to spread out computational and storage workloads across leader nodes. One of our future works will fully incorporate the sharding approach in both Fog and Cloud layer.

5. The work in this thesis described how the Patient Agent would fit in a 5G network and manage resources demand from diverse health applications. However, the Patient Agent was not implemented on a 5G network. Future work is to incorporate the Patient Agent in 5G network and analyze the performances by implementing a prototype of the framework at the hardware.

Bibliography

[1] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Integration of blockchain and cloud of things: Architecture, applications and challenges," *arXiv preprint arXiv:1908.09058*, 2019.

[2] S. S. Panda, U. Satapathy, B. K. Mohanta, D. Jena, and D. Gountia, "A blockchain based decentralized authentication framework for resource constrained iot devices," in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE, 2019, pp. 1–6.

[3] M. A. Khan and K. Salah, "Iot security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems*, vol. 82, pp. 395–411, 2018.

[4] N. Siegfried, T. Rosenthal, A. Benlian *et al.*, "Blockchain and the industrial internet of things: A requirement taxonomy and systematic fit analysis," Darmstadt Technical University, Department of Business Administration . . . , Tech. Rep., 2020.

[5] R. A. Michelin, A. Dorri, M. Steger, R. C. Lunardi, S. S. Kanhere, R. Jurdak, and A. F. Zorzo, "Speedychain: A framework for decoupling data from blockchain for smart cities," in *Proceedings of the 15th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2018, pp. 145–154.

[6] Y. Yu, Y. Li, J. Tian, and J. Liu, "Blockchain-based solutions to security and privacy issues in the internet of things," *IEEE Wireless Communications*, vol. 25, no. 6, pp. 12–18, 2018.

[7] A. Panarello, N. Tapas, G. Merlino, F. Longo, and A. Puliafito, "Blockchain and iot integration: A systematic survey," *Sensors*, vol. 18, no. 8, p. 2575, 2018.

[8] J. Huang, L. Kong, G. Chen, M.-Y. Wu, X. Liu, and P. Zeng, "Towards secure industrial iot: Blockchain system with credit-based consensus mechanism," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3680–3689, 2019.

[9] Q. Zhou, H. Huang, Z. Zheng, and J. Bian, "Solutions to scalability of blockchain: A survey," *IEEE Access*, vol. 8, pp. 16 440–16 455, 2020.

[10] P. P. Ray, D. Dash, K. Salah, and N. Kumar, "Blockchain for iot-based healthcare: Background, consensus, platforms, and use cases," *IEEE Systems Journal*, 2020.

[11] P. De Filippi, M. Mannan, and W. Reijers, "Blockchain as a confidence machine: The problem of trust & challenges of governance," *Technology in Society*, vol. 62, p. 101284, 2020.

[12] A. Antonopoulos, "Bitcoin security model: trust by computation," *O'Reilly Radar. Retrieved October*, vol. 4, p. 2015, 2014.

[13] K. R. Özyılmaz and A. Yurdakul, "Work-in-progress: integrating low-power iot devices to a blockchain-based infrastructure," in *2017 International Conference on Embedded Software (EMSOFT)*. IEEE, 2017, pp. 1–2.

[14] K. J. O'Dwyer and D. Malone, "Bitcoin mining and its energy footprint," 2014.

[15] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasubramanian, "An efficient selective miner consensus protocol in blockchain oriented iot smart monitoring." in *ICIT*, 2019, pp. 1135–1142.

[16] A. Sharma, S. Bahl, A. K. Bagha, M. Javaid, D. K. Shukla, and A. Haleem, "Blockchain technology and its applications to combat covid-19 pandemic," *Research on Biomedical Engineering*, pp. 1–8, 2020.

[17] A. D. Dwivedi, L. Malina, P. Dzurenda, and G. Srivastava, "Optimized blockchain model for internet of things based healthcare applications," in *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*. IEEE, 2019, pp. 135–139.

[18] H. F. Atlam and G. B. Wills, "Technical aspects of blockchain and iot," in *Advances in Computers*. Elsevier, 2019, vol. 115, pp. 1–39.

[19] E. Karafiloski and A. Mishev, "Blockchain solutions for big data challenges: A literature review," in *IEEE EUROCON 2017-17th International Conference on Smart Technologies*. IEEE, 2017, pp. 763–768.

[20] "Blockchain issues: 1: Data storage, retrieved from https://medium.com/@kyle.may/blockchain-issues-1-data-storage."

[21] T. Yu, X. Wang, and Y. Zhu, "Blockchain technology for the 5g-enabled internet of things systems: Principle, applications and challenges," *5G-Enabled Internet of Things*, 2019.

[22] J. Ellul, J. Galea, M. Ganado, S. Mccarthy, and G. J. Pace, "Regulating blockchain, dlt and smart contracts: a technology regulator's perspective," in *ERA Forum*, vol. 21, no. 2. Springer, 2020, pp. 209–220.

[23] A. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz, "On blockchain and its integration with iot. challenges and opportunities," *Future Generation Computer Systems*, vol. 88, pp. 173–190, 2018.

[24] J. Kang, Z. Xiong, D. Niyato, D. Ye, D. I. Kim, and J. Zhao, "Toward secure blockchain-enabled internet of vehicles: Optimizing consensus management using reputation and contract theory," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2906–2920, 2019.

[25] A. F. Zorzo, H. C. Nunes, R. C. Lunardi, R. A. Michelin, and S. S. Kanhere, "Dependable iot using blockchain-based technology," in *2018 Eighth Latin-American Symposium on Dependable Computing (LADC)*. IEEE, 2018, pp. 1–9.

[26] R. P. Naik and N. T. Courtois, "Optimising the sha256 hashing algorithm for faster and more efficient bitcoin mining," *MSc Information Security Department of Computer Science UCL*, pp. 1–65, 2013.

[27] A. Firdaus, M. F. Ab Razak, A. Feizollah, I. A. T. Hashem, M. Hazim, and N. B. Anuar, "The rise of "blockchain": bibliometric analysis of blockchain study," *Scientometrics*, vol. 120, no. 3, pp. 1289–1331, 2019.

[28] B. Yuan, H. Jin, D. Zou, L. T. Yang, and S. Yu, "A practical byzantine-based approach for faulty switch tolerance in software-defined networks," *IEEE Transactions on Network and Service Management*, vol. 15, no. 2, pp. 825–839, 2018.

[29] L. Ismail and H. Materwala, "A review of blockchain architecture and consensus protocols: Use cases, challenges, and solutions," *Symmetry*, vol. 11, no. 10, p. 1198, 2019.

[30] V. Gramoli, "From blockchain consensus back to byzantine consensus," *Future Generation Computer Systems*, 2017.

[31] A. Corso, "Performance analysis of proof-of-elapsed-time (poet) consensus in the sawtooth blockchain framework," 2019.

[32] S. Biswas, K. Sharif, F. Li, S. Maharjan, S. P. Mohanty, and Y. Wang, "Pobt: A lightweight consensus algorithm for scalable iot business blockchain," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 2343–2355, 2019.

[33] S. Hakak, W. Z. Khan, G. A. Gilkar, M. Imran, and N. Guizani, "Securing smart cities through blockchain technology: Architecture, requirements, and challenges," *IEEE Network*, vol. 34, no. 1, pp. 8–14, 2020.

[34] F. Yang, W. Zhou, Q. Wu, R. Long, N. N. Xiong, and M. Zhou, "Delegated proof of stake with downgrade: A secure and efficient blockchain consensus algorithm with downgrade mechanism," *IEEE Access*, vol. 7, pp. 118 541–118 555, 2019.

[35] 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, p. 130, 2018.

[36] J. Pan, J. Wang, A. Hester, I. Alqerm, Y. Liu, and Y. Zhao, "Edgechain: An edge-iot framework and prototype based on blockchain and smart contracts," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4719–4732, 2018.

[37] J. Frizzo-Barker, P. A. Chow-White, P. R. Adams, J. Mentanko, D. Ha, and S. Green, "Blockchain as a disruptive technology for business: A systematic review," *International Journal of Information Management*, 2019.

[38] Y. Hu, M. Liyanage, A. Mansoor, K. Thilakarathna, G. Jourjon, and A. Seneviratne, "Blockchain-based smart contracts-applications and challenges," *arXiv preprint arXiv:1810.04699*, 2018.

[39] U. Bodkhe, S. Tanwar, K. Parekh, P. Khanpara, S. Tyagi, N. Kumar, and M. Alazab, "Blockchain for industry 4.0: A comprehensive review," *IEEE Access*, vol. 8, pp. 79 764–79 800, 2020.

[40] S. Tanwar, K. Parekh, and R. Evans, "Blockchain-based electronic healthcare record system for healthcare 4.0 applications," *Journal of Information Security and Applications*, vol. 50, p. 102407, 2020.

[41] F. Lin and M. Qiang, "The challenges of existence, status, and value for improving blockchain," *IEEE Access*, vol. 7, pp. 7747–7758, 2018.

[42] S. Kim, Y. Kwon, and S. Cho, "A survey of scalability solutions on blockchain," in *2018 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2018, pp. 1204–1207.

[43] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum project yellow paper*, vol. 151, pp. 1–32, 2014.

[44] A. Jabbar and S. Dani, "Investigating the link between transaction and computational costs in a blockchain environment," *International Journal of Production Research*, vol. 58, no. 11, pp. 3423–3436, 2020.

[45] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao, "Sybillimit: A near-optimal social network defense against sybil attacks," in *2008 IEEE Symposium on Security and Privacy (sp 2008)*. IEEE, 2008, pp. 3–17.

[46] R. J. Tom, S. Sankaranarayanan, and J. J. Rodrigues, "Agent negotiation in an iot-fog based power distribution system for demand reduction," *Sustainable Energy Technologies and Assessments*, vol. 38, p. 100653, 2020.

[47] D. Calvaresi, A. Dubovitskaya, J. P. Calbimonte, K. Taveter, and M. Schumacher, "Multi-agent systems and blockchain: Results from a systematic literature review," in *International Conference on Practical Applications of Agents and Multi-Agent Systems*. Springer, 2018, pp. 110–126.

[48] K. Qayumi, "Multi-agent based intelligence generation from very large datasets," in *2015 IEEE International Conference on Cloud Engineering*. IEEE, 2015, pp. 502–504.

[49] F. Luo, Z. Y. Dong, G. Liang, J. Murata, and Z. Xu, "A distributed electricity trading system in active distribution networks based on multi-agent coalition and blockchain," *IEEE Transactions on Power Systems*, vol. 34, no. 5, pp. 4097–4108, 2018.

[50] J. K. M. Verame, "Helping users adopt and delegate agency to autonomous agents in everyday life," Ph.D. dissertation, University of Southampton, 2018.

[51] A. Norta, A. B. Othman, and K. Taveter, "Conflict-resolution lifecycles for governed decentralized autonomous organization collaboration," in *Proceedings of the 2015 2nd International Conference on Electronic Governance and Open Society: Challenges in Eurasia*, 2015, pp. 244–257.

[52] P. De Meo, F. Messina, D. Rosaci, and G. M. Sarné, "Recommending users in social networks by integrating local and global reputation," in *International Conference on Internet and Distributed Computing Systems*. Springer, 2014, pp. 437–446.

[53] X. Zheng, R. R. Mukkamala, R. Vatrapu, and J. Ordieres-Mere, "Blockchain-based personal health data sharing system using cloud storage," in *2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom)*. IEEE, 2018, pp. 1–6.

[54] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasubramanian, "Continuous patient monitoring with a patient centric agent: A block architecture," *IEEE Access*, vol. 6, pp. 32 700–32 726, 2018.

[55] G. Fortino, F. Messina, D. Rosaci, and G. M. Sarne, "Using blockchain in a reputation-based model for grouping agents in the internet of things," *IEEE Transactions on Engineering Management*, 2019.

[56] S.-C. Cha, J.-F. Chen, C. Su, and K.-H. Yeh, "A blockchain connected gateway for ble-based devices in the internet of things," *IEEE Access*, vol. 6, pp. 24 639–24 649, 2018.

[57] S. R. Islam, D. Kwak, M. H. Kabir, M. Hossain, and K.-S. Kwak, "The internet of things for health care: a comprehensive survey," *IEEE Access*, vol. 3, pp. 678–708, 2015.

[58] L. Wang, G.-Z. Yang, J. Huang, J. Zhang, L. Yu, Z. Nie, and D. R. S. Cumming, "A wireless biomedical signal interface system-on-chip for body sensor networks," *IEEE Transactions on biomedical circuits and systems*, vol. 4, no. 2, pp. 112–117, 2010.

[59] A. Marrington, D. Kerr, and J. Gammack, *Managing Security Issues and the Hidden Dangers of Wearable Technologies*. IGI Global, 2016.

[60] A. Chib, M. H. van Velthoven, and J. Car, "mhealth adoption in low-resource environments: a review of the use of mobile healthcare in developing countries," *Journal of health communication*, vol. 20, no. 1, pp. 4–34, 2015.

[61] S. S. Khanuja, S. Garg, and I. P. Singh, "Method and apparatus for remotely monitoring the condition of a patient," May 7 2009, uS Patent App. 12/259,905.

[62] A. Vegesna, M. Tran, M. Angelaccio, and S. Arcona, "Remote patient monitoring via non-invasive digital technologies: a systematic review," *Telemedicine and e-Health*, vol. 23, no. 1, pp. 3–17, 2017.

[63] B. Séroussi and J. Bouaud, "Adoption of a nationwide shared medical record in france: Lessons learnt after 5 years of deployment," in *AMIA Annual Symposium Proceedings*, vol. 2016. American Medical Informatics Association, 2016, p. 1100.

[64] A. Garavand, M. Samadbeik, M. Kafashi, and S. Abhari, "The identification and classification of deployment challenges related to electronic health records: A review article," *Shiraz E-Medical Journal*, vol. 17, no. 2, 2016.

[65] J. Allen-Graham, L. Mitchell, N. Heriot, R. Armani, D. Langton, M. Levinson, A. Young, J. A. Smith, T. Kotsimbos, and J. W. Wilson, "Electronic health records and online medical records: an asset or a liability under current conditions?" *Australian Health Review*, vol. 42, no. 1, pp. 59–65, 2018.

[66] J. D. Halamka and A. Ekblaw, "The potential for blockchain to transform electronic health records," *Harvard Business Review*, vol. 3, 2017.

[67] M. Mosmondor, I. Benc, S. Desic, and A. Grguric, "A feasibility study for the integration of a remote patient monitoring solution with electronic health record system," in *The 33rd International Convention MIPRO*. IEEE, 2010, pp. 360–366.

[68] V. Balasubramanian, A. Stranieri, and R. Kaur, "Appa: assistive patient monitoring cloud platform for active healthcare applications," in *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication*. ACM, 2015, p. 54.

[69] W. Ni, X. Huang, J. Zhang, and R. Yu, "Healchain: A decentralized data management system for mobile healthcare using consortium blockchain," in *2019 Chinese Control Conference (CCC)*. IEEE, 2019, pp. 6333–6338.

[70] R. Spiegel, "Manufacturing is more vulnerable to cyber attack as iot proliferates," *Available from https://www.designnews.com/automotive-0, Accessed: 11 May, 2018*, 2016.

[71] L. Blake, V. Francis, J. Johnson, M. Khan, and T. McCray, "Developing robust data management strategies for unprecedented challenges to healthcare information," *Journal of Leadership, Accountability and Ethics*, vol. 14, no. 1, p. 22, 2017.

[72] M. Niranjanamurthy, B. Nithya, and S. Jagannatha, "Analysis of blockchain technology: pros, cons and swot," *Cluster Computing*, pp. 1–15.

[73] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[74] V. Buterin *et al.*, "A next-generation smart contract and decentralized application platform," *white paper*, 2014.

[75] Q. Xia, E. B. Sifah, K. O. Asamoah, J. Gao, X. Du, and M. Guizani, "Medshare: Trust-less medical data sharing among cloud service providers via blockchain," *IEEE Access*, 2017.

[76] J. Zhang, N. Xue, and X. Huang, "A secure system for pervasive social network-based healthcare," *IEEE Access*, vol. 4, pp. 9239–9250, 2016.

[77] 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, p. 218, 2016.

[78] B. Health, "The future of digital health," *Retrieved from https://www.bowheadhealth.com, Accessed 9 September, 2017*, 2017.

[79] H. Zhao, Y. Zhang, Y. Peng, and R. Xu, "Lightweight backup and efficient recovery scheme for health blockchain keys," in *Autonomous Decentralized System (ISADS), 2017 IEEE 13th International Symposium on*. IEEE, 2017, pp. 229–234.

[80] L. A. Linn and M. B. Koo, "Blockchain for health data and its potential use in health it and health care related research," in *ONC/NIST Use of Blockchain for Healthcare and Research Workshop. Gaithersburg, Maryland, United States: ONC/NIST*, 2016.

[81] J. Eberhardt and J. Heiss, "Off-chaining models and approaches to off-chain computations," in *Proceedings of the 2nd Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers*, 2018, pp. 7–12.

[82] B. Cao, Y. Li, L. Zhang, L. Zhang, S. Mumtaz, Z. Zhou, and M. Peng, "When internet of things meets blockchain: Challenges in distributed consensus," *IEEE Network*, 2019.

[83] M. M. H. Onik, S. Aich, J. Yang, C.-S. Kim, and H.-C. Kim, "Blockchain in healthcare: Challenges and solutions," in *Big Data Analytics for Intelligent Healthcare Management*. Elsevier, 2019, pp. 197–226.

[84] P. Zhang, J. White, D. C. Schmidt, G. Lenz, and S. T. Rosenbloom, "Fhirchain: applying blockchain to securely and scalably share clinical data," *Computational and structural biotechnology journal*, vol. 16, pp. 267–278, 2018.

[85] M. Aazam and E.-N. Huh, "Dynamic resource provisioning through fog micro datacenter," in *Pervasive Computing and Communication Workshops (PerCom Workshops), 2015 IEEE International Conference on*. IEEE, 2015, pp. 105–110.

[86] E. Baccarelli, P. G. V. Naranjo, M. Scarpiniti, M. Shojafar, and J. H. Abawajy, "Fog of everything: Energy-efficient networked computing architectures, research challenges, and a case study," *IEEE access*, vol. 5, pp. 9882–9910, 2017.

[87] R. Mahmud, F. L. Koch, and R. Buyya, "Cloud-fog interoperability in iot-enabled healthcare solutions," in *Proceedings of the 19th international conference on distributed computing and networking*, 2018, pp. 1–10.

[88] A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang, and P. Liljeberg, "Exploiting smart e-health gateways at the edge of healthcare internet-of-things: a fog computing approach," *Future Generation Computer Systems*, vol. 78, pp. 641–658, 2018.

[89] M. A. Uddin, A. Stranieri, I. Gondal, and Balasubramanian, "A patient agent to manage blockchains for remote patient monitoring," *Studies in health technology and informatics*, vol. 254, pp. 105–115, 2018.

[90] S. Tuli, R. Mahmud, S. Tuli, and R. Buyya, "Fogbus: A blockchain-based lightweight framework for edge and fog computing," *arXiv preprint arXiv:1811.11978*, 2018.

[91] 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.

[92] S. Kitanov and T. Janevski, "Introduction to fog computing," *The Rise of Fog Computing in the Digital Era*, p. 1, 2018.

[93] R. Fang, S. Pouyanfar, Y. Yang, S.-C. Chen, and S. Iyengar, "Computational health informatics in the big data age: a survey," *ACM Computing Surveys (CSUR)*, vol. 49, no. 1, p. 12, 2016.

[94] A. Stranieri and V. Balasubramanian, "Remote patient monitoring for healthcare: A big challenge for big data," in *Managerial Perspectives on Intelligent Big Data Analytics*. IGI Global, 2019, pp. 163–179.

[95] P. Bhattacharya, S. Tanwar, U. Bodke, S. Tyagi, and N. Kumar, "Bindaas: Blockchain-based deep-learning as-a-service in healthcare 4.0 applications," *IEEE Transactions on Network Science and Engineering*, 2019.

[96] "Fit for everybody," *Fitbit Official Site for Activity Trackers and More, Accessed from https://www.fitbit.com/au/home*.

[97] S. Lee and K.-K. Seo, "A hybrid multi-criteria decision-making model for a cloud service selection problem using bsc, fuzzy delphi method and fuzzy ahp," *Wireless Personal Communications*, vol. 86, no. 1, pp. 57–75, 2016.

[98] S. Alismaili, M. Li, and J. Shen, "Cloud computing adoption decision modelling for smes: from the paprika perspective," in *Frontier Computing*. Springer, 2016, pp. 597–615.

[99] M. C. Domingo, "An overview of the internet of underwater things," *Journal of Network and Computer Applications*, vol. 35, no. 6, pp. 1879–1890, 2012.

[100] A. Akhter, M. A. Uddin, M. A. I. Abir, and M. M. Islam, "Noise aware level based routing protocol for underwater sensor networks," in *2016 International Workshop on Computational Intelligence (IWCI)*. IEEE, 2016, pp. 169–174.

[101] N. Li, J.-F. Martínez, J. Meneses Chaus, and M. Eckert, "A survey on underwater acoustic sensor network routing protocols," *Sensors*, vol. 16, no. 3, p. 414, 2016.

[102] G. Han, J. Jiang, N. Sun, and L. Shu, "Secure communication for underwater acoustic sensor networks," *IEEE communications magazine*, vol. 53, no. 8, pp. 54–60, 2015.

[103] I. F. Akyildiz, D. Pompili, and T. Melodia, "Underwater acoustic sensor networks: research challenges," *Ad hoc networks*, vol. 3, no. 3, pp. 257–279, 2005.

[104] M. Uddin *et al.*, "Link expiration time-aware routing protocol for uwsns," *Journal of Sensors*, vol. 2013, 2013.

[105] M. Faheem, G. Tuna, and V. C. Gungor, "Lrp: Link quality-aware queue-based spectral clustering routing protocol for underwater acoustic sensor networks," *International Journal of Communication Systems*, vol. 30, no. 12, p. e3257, 2017.

[106] M. Kim, K.-S. Lim, J. Song, and M.-s. Jun, "An efficient secure scheme based on hierarchical topology in the smart home environment," *Symmetry*, vol. 9, no. 8, p. 143, 2017.

[107] B. Jo, R. Khan, and Y.-S. Lee, "Hybrid blockchain and internet-of-things network for underground structure health monitoring," *Sensors*, vol. 18, no. 12, p. 4268, 2018.

[108] J. Jiang, G. Han, H. Guo, L. Shu, and J. J. Rodrigues, "Geographic multipath routing based on geospatial division in duty-cycled underwater wireless sensor networks," *Journal of Network and Computer Applications*, vol. 59, pp. 4–13, 2016.

[109] F. Al Salti, N. Alzeidi, and B. R. Arafeh, "Emggr: an energy-efficient multipath grid-based geographic routing protocol for underwater wireless sensor networks," *Wireless Networks*, vol. 23, no. 4, pp. 1301–1314, 2017.

[110] Z. Wei, M. Song, G. Yin, H. Song, H. Wang, X. Ma, and A. Cheng, "Data access based on a guide map of the underwater wireless sensor network," *Sensors*, vol. 17, no. 10, p. 2374, 2017.

[111] C. Lal, R. Petroccia, K. Pelekanakis, M. Conti, and J. Alves, "Toward the development of secure underwater acoustic networks," *IEEE Journal of Oceanic Engineering*, vol. 42, no. 4, pp. 1075–1087, 2017.

[112] A. Dhumane, R. Prasad, and J. Prasad, "Routing issues in internet of things: a survey," in *Proceedings of the international multiconference of engineers and computer scientists*, vol. 1, 2016, pp. 16–18.

[113] G. Liu and C. Wei, "A new multi-path routing protocol based on cluster for underwater acoustic sensor networks," in *2011 International Conference on Multimedia Technology*. IEEE, 2011, pp. 91–94.

[114] L. Guangzhong and L. Zhibin, "Depth-based multi-hop routing protocol for underwater sensor network," in *2010 The 2nd International Conference on Industrial Mechatronics and Automation*, vol. 2. IEEE, 2010, pp. 268–270.

[115] A. Wahid, S. Lee, and D. Kim, "A reliable and energy-efficient routing protocol for underwater wireless sensor networks," *International Journal of Communication Systems*, vol. 27, no. 10, pp. 2048–2062, 2014.

[116] M. C. Domingo, "A distributed energy-aware routing protocol for underwater wireless sensor networks," *Wireless Personal Communications*, vol. 57, no. 4, pp. 607–627, 2011.

[117] M. Ayaz, A. Abdullah, and L. T. Jung, "Temporary cluster based routing for underwater wireless sensor networks," in *2010 International Symposium on Information Technology*, vol. 2. IEEE, 2010, pp. 1009–1014.

[118] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[119] I. Jawhar, N. Mohamed, J. Al-Jaroodi, and S. Zhang, "An architecture for using autonomous underwater vehicles in wireless sensor networks for underwater pipeline monitoring," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 3, pp. 1329–1340, 2018.

[120] N. Mohamed, I. Jawhar, J. Al-Jaroodi, and L. Zhang, "Sensor network architectures for monitoring underwater pipelines," *Sensors*, vol. 11, no. 11, pp. 10 738–10 764, 2011.

[121] T. M. Fernández-Caramés and P. Fraga-Lamas, "A review on the use of blockchain for the internet of things," *IEEE Access*, vol. 6, pp. 32 979–33 001, 2018.

[122] A. Yazdinejad, R. M. Parizi, G. Srivastava, A. Dehghantanha, and K.-K. R. Choo, "Energy efficient decentralized authentication in internet of underwater things using blockchain," in *2019 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2019, pp. 1–6.

[123] Y. Lee, S. Rathore, J. H. Park, and J. H. Park, "A blockchain-based smart home gateway architecture for preventing data forgery," *Human-centric Computing and Information Sciences*, vol. 10, no. 1, pp. 1–14, 2020.

[124] M. Schiefer, "Smart home definition and security threats," in *2015 ninth international conference on IT security incident management & IT forensics*. IEEE, 2015, pp. 114–118.

[125] K. Gu, L. Yang, and B. Yin, "Location data record privacy protection based on differential privacy mechanism." in *ITC*, vol. 47, no. 4, 2018, pp. 639–654.

[126] P. K. Sharma, S. Rathore, and J. H. Park, "Distarch-scnet: blockchain-based distributed architecture with li-fi communication for a scalable smart city network," *IEEE Consumer Electronics Magazine*, vol. 7, no. 4, pp. 55–64, 2018.

[127] S. Tuli, S. Tuli, G. Wander, P. Wander, S. S. Gill, S. Dustdar, R. Sakellariou, and O. Rana, "Next generation technologies for smart healthcare: Challenges, vision, model, trends and future directions," *Internet Technology Letters*, p. e145, 2019.

[128] H. Gupta, A. Vahid Dastjerdi, S. K. Ghosh, and R. Buyya, "ifogsim: A toolkit for modeling and simulation of resource management techniques in the internet of things, edge and fog computing environments," *Software: Practice and Experience*, vol. 47, no. 9, pp. 1275–1296, 2017.

[129] C. J. Cremers, "The scyther tool: Verification, falsification, and analysis of security protocols," in *International Conference on Computer Aided Verification*. Springer, 2008, pp. 414–418.

[130] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *Computational Intelligence for Security and Defense Applications, 2009. CISDA 2009. IEEE Symposium on*. IEEE, 2009, pp. 1–6.

[131] A. Noureddine, S. Islam, and R. Bashroush, "Jolinar: analysing the energy footprint of software applications," in *Proceedings of the 25th International Symposium on Software Testing and Analysis*. ACM, 2016, pp. 445–448.

[132] F. M. Benčić and I. P. Žarko, "Distributed ledger technology: Blockchain compared to directed acyclic graph," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2018, pp. 1569–1570.

[133] V. Acharya, A. E. Yerrapati, and N. Prakash, *Oracle Blockchain Quick Start Guide: A practical approach to implementing blockchain in your enterprise*. Packt Publishing Ltd, 2019.

[134] C. Fan, S. Ghaemi, H. Khazaei, and P. Musilek, "Performance evaluation of blockchain systems: A systematic survey," *IEEE Access*, vol. 8, pp. 126 927–126 950, 2020.

[135] T. Alladi, V. Chamola, R. M. Parizi, and K.-K. R. Choo, "Blockchain applications for industry 4.0 and industrial iot: A review," *IEEE Access*, vol. 7, pp. 176 935–176 951, 2019.

[136] L. Wang, X. Shen, J. Li, J. Shao, and Y. Yang, "Cryptographic primitives in blockchains," *Journal of Network and Computer Applications*, vol. 127, pp. 43–58, 2019.

[137] D. Boneh, *Aggregate Signatures*. Boston, MA: Springer US, 2011, pp. 27–27. [Online]. Available: https://doi.org/10.1007/978-1-4419-5906-5_139

[138] W. Fang, W. Chen, W. Zhang, J. Pei, W. Gao, and G. Wang, "Digital signature scheme for information non-repudiation in blockchain: a state of the art review," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, pp. 1–15, 2020.

[139] K. Huang, X. Zhang, Y. Mu, F. Rezaeibagha, and X. Du, "Scalable and redactable blockchain with update and anonymity," *Information Sciences*, vol. 546, pp. 25–41.

[140] C. Li, Y. Tian, X. Chen, and J. Li, "An efficient anti-quantum lattice-based blind signature for blockchain-enabled systems," *Information Sciences*, vol. 546, pp. 253–264, 2020.

[141] A. Manzoor, A. Braeken, S. S. Kanhere, M. Ylianttila, and M. Liyanage, "Proxy re-encryption enabled secure and anonymous iot data sharing platform based on blockchain," *Journal of Network and Computer Applications*, p. 102917, 2020.

[142] L. Peng, W. Feng, Z. Yan, Y. Li, X. Zhou, and S. Shimizu, "Privacy preservation in permissionless blockchain: A survey," *Digital Communications and Networks*, 2020.

[143] A. R. Taleb and D. Vergnaud, "Speeding-up verification of digital signatures," *Journal of Computer and System Sciences*, vol. 116, pp. 22–39, 2020.

[144] Z. Wang, H. Yu, Z. Zhang, J. Piao, and J. Liu, "Ecdsa weak randomness in bitcoin," *Future Generation Computer Systems*, vol. 102, pp. 507–513, 2020.

[145] Q. Zhao, S. Chen, Z. Liu, T. Baker, and Y. Zhang, "Blockchain-based privacy-preserving remote data integrity checking scheme for iot information systems," *Information Processing & Management*, vol. 57, no. 6, p. 102355, 2020.

[146] C.-F. Chou, W. C. Cheng, and L. Golubchik, "Performance study of online batch-based digital signature schemes," *Journal of Network and Computer Applications*, vol. 33, no. 2, pp. 98–114, 2010.

[147] M. Michels, D. Naccache, H. Petersen *et al.*, "Gost 34.10-a brief overview of russia's dsa," *Computers and Security*, vol. 15, no. 8, pp. 725–732, 1996.

[148] H. Morita, J. C. Schuldt, T. Matsuda, G. Hanaoka, and T. Iwata, "On the security of the schnorr signature scheme and dsa against related-key attacks," in *ICISC 2015*. Springer, 2015, pp. 20–35.

[149] Z. Lyasota, "A guide to digital signature algorithms-dzone security," Aug 2018. [Online]. Available: https://dzone.com/articles/digital-signature-1

[150] M. Elia, M. Piva, and D. Schipani, "The rabin cryptosystem revisited," *Applicable Algebra in Engineering, Communication and Computing*, vol. 26, no. 3, pp. 251–275, 2015.

[151] M. Ghosh, M. Richardson, B. Ford, and R. Jansen, "A torpath to torcoin: Proof-of-bandwidth altcoins for compensating relays," NAVAL RESEARCH LAB WASHINGTON DC, Tech. Rep., 2014.

[152] S. De Angelis, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone, "Pbft vs proof-of-authority: Applying the cap theorem to permissioned blockchain," 2018.

[153] D. Puthal and S. P. Mohanty, "Proof of authentication: Iot-friendly blockchains," *IEEE Potentials*, vol. 38, no. 1, pp. 26–29, 2018.

[154] D. Puthal, S. P. Mohanty, P. Nanda, E. Kougianos, and G. Das, "Proof-of-authentication for scalable blockchain in resource-constrained distributed systems," in *2019 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 2019, pp. 1–5.

[155] G. Yu, X. Wang, K. Yu, W. Ni, J. A. Zhang, and R. P. Liu, "Survey: Sharding in blockchains," *IEEE Access*, vol. 8, pp. 14 155–14 181, 2020.

[156] A. Hafid, A. S. Hafid, and M. Samih, "Scaling blockchains: A comprehensive survey," *IEEE Access*, vol. 8, pp. 125 244–125 262, 2020.

[157] H. Chen and Y. Wang, "Sschain: A full sharding protocol for public blockchain without data migration overhead," *Pervasive and Mobile Computing*, vol. 59, p. 101055, 2019.

[158] T. Hewa, M. Ylianttila, and M. Liyanage, "Survey on blockchain based smart contracts: Applications, opportunities and challenges," *Journal of Network and Computer Applications*, p. 102857, 2020.

[159] F. Jamil, S. Ahmad, N. Iqbal, and D.-H. Kim, "Towards a remote monitoring of patient vital signs based on iot-based blockchain integrity management platforms in smart hospitals," *Sensors*, vol. 20, no. 8, p. 2195, 2020.

[160] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Blockchain for secure ehrs sharing of mobile cloud based e-health systems," *IEEE access*, vol. 7, pp. 66 792–66 806, 2019.

[161] S. Wang, Y. Zhang, and Y. Zhang, "A blockchain-based framework for data sharing with fine-grained access control in decentralized storage systems," *IEEE Access*, vol. 6, pp. 38 437–38 450, 2018.

[162] R. Akkaoui, X. Hei, and W. Cheng, "Edgemedichain: A hybrid edge blockchain-based framework for health data exchange," *IEEE Access*, vol. 8, pp. 113 467–113 486, 2020.

[163] R. C. Celiz, Y. E. De La Cruz, and D. M. Sanchez, "Cloud model for purchase management in health sector of peru based on iot and blockchain," in *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*. IEEE, 2018, pp. 328–334.

[164] J. Liu, X. Li, L. Ye, H. Zhang, X. Du, and M. Guizani, "Bpds: A blockchain based privacy-preserving data sharing for electronic medical records," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 1–6.

[165] H. Kaur, M. A. Alam, R. Jameel, A. K. Mourya, and V. Chang, "A proposed solution and future direction for blockchain-based heterogeneous medicare data in cloud environment," *Journal of medical systems*, vol. 42, no. 8, p. 156, 2018.

[166] A. Al Omar, M. Z. A. Bhuiyan, A. Basu, S. Kiyomoto, and M. S. Rahman, "Privacy-friendly platform for healthcare data in cloud based on blockchain environment," *Future Generation Computer Systems*, vol. 95, pp. 511–521, 2019.

[167] L. Hang, E. Choi, and D.-H. Kim, "A novel emr integrity management based on a medical blockchain platform in hospital," *Electronics*, vol. 8, no. 4, p. 467, 2019.

[168] S. Cao, G. Zhang, P. Liu, X. Zhang, and F. Neri, "Cloud-assisted secure ehealth systems for tamper-proofing ehr via blockchain," *Information Sciences*, vol. 485, pp. 427–440, 2019.

[169] J. Park, S. Park, K. Kim, and D. Lee, "Corus: Blockchain-based trustworthy evaluation system for efficacy of healthcare remedies," in *2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*. IEEE, 2018, pp. 181–184.

[170] E.-Y. Daraghmi, Y.-A. Daraghmi, and S.-M. Yuan, "Medchain: a design of blockchain-based system for medical records access and permissions management," *IEEE Access*, vol. 7, pp. 164 595–164 613, 2019.

[171] Y. Chen, S. Ding, Z. Xu, H. Zheng, and S. Yang, "Blockchain-based medical records secure storage and medical service framework," *Journal of medical systems*, vol. 43, no. 1, p. 5, 2019.

[172] M. A. Rahman, M. M. Rashid, M. S. Hossain, E. Hassanain, M. F. Alhamid, and M. Guizani, "Blockchain and iot-based cognitive edge framework for sharing economy services in a smart city," *IEEE Access*, vol. 7, pp. 18 611–18 621, 2019.

[173] A. Manzoor, M. Liyanage, A. Braeke, S. S. Kanhere, and M. Ylianttila, "Blockchain based proxy re-encryption scheme for secure iot data sharing," in *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. IEEE, 2019, pp. 99–103.

[174] M. Debe, K. Salah, M. H. U. Rehman, and D. Svetinovic, "Monetization of services provided by public fog nodes using blockchain and smart contracts," *IEEE Access*, vol. 8, pp. 20 118–20 128, 2020.

[175] H. S. Z. Kazmi, F. Nazeer, S. Mubarak, S. Hameed, A. Basharat, and N. Javaid, "Trusted remote patient monitoring using blockchain-based smart contracts," in *International Conference on Broadband and Wireless Computing, Communication and Applications*. Springer, 2019, pp. 765–776.

[176] V. Malamas, T. Dasaklis, P. Kotzanikolaou, M. Burmester, and S. Katsikas, "A forensics-by-design management framework for medical devices based on blockchain," in *2019 IEEE World Congress on Services (SERVICES)*, vol. 2642. IEEE, 2019, pp. 35–40.

[177] Y. Xu, G. Wang, J. Yang, J. Ren, Y. Zhang, and C. Zhang, "Towards secure network computing services for lightweight clients using blockchain," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

[178] R. Almadhoun, M. Kadadha, M. Alhemeiri, M. Alshehhi, and K. Salah, "A user authentication scheme of iot devices using blockchain-enabled fog nodes," in *2018 IEEE/ACS 15th international conference on computer systems and applications (AICCSA)*. IEEE, 2018, pp. 1–8.

[179] T. D. Nguyen, H.-A. Pham, and M. T. Thai, "Leveraging blockchain to enhance data privacy in iot-based applications," in *International Conference on Computational Social Networks*. Springer, 2018, pp. 211–221.

[180] P. Mytis-Gkometh, G. Drosatos, P. Efraimidis, and E. Kaldoudi, "Notarization of knowledge retrieval from biomedical repositories using blockchain technology," in *International Conference on Biomedical and Health Informatics*. Springer, 2017, pp. 69–73.

[181] H. Yu, Z. Yang, and R. O. Sinnott, "Decentralized big data auditing for smart city environments leveraging blockchain technology," *IEEE Access*, vol. 7, pp. 6288–6296, 2018.

[182] M. Baza, M. Nabil, N. Lasla, K. Fidan, M. Mahmoud, and M. Abdallah, "Blockchain-based firmware update scheme tailored for autonomous vehicles," in *2019 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2019, pp. 1–7.

[183] S. Malik, V. Dedeoglu, S. S. Kanhere, and R. Jurdak, "Trustchain: Trust management in blockchain and iot supported supply chains," in *2019 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 2019, pp. 184–193.

[184] D. Calvaresi, V. Mattioli, A. Dubovitskaya, A. F. Dragoni, and M. Schumacher, "Reputation management in multi-agent systems using permissioned blockchain technology," in *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. IEEE, 2018, pp. 719–725.

[185] M. Debe, K. Salah, M. H. U. Rehman, and D. Svetinovic, "Iot public fog nodes reputation system: A decentralized solution using ethereum blockchain," *IEEE Access*, vol. 7, pp. 178 082–178 093, 2019.

[186] S. El Kafhali, C. Chahir, M. Hanini, and K. Salah, "Architecture to manage internet of things data using blockchain and fog computing," in *Proceedings of the 4th International Conference on Big Data and Internet of Things*, 2019, pp. 1–8.

[187] S. Talukder, S. Roy, and T. Al Mahmud, "A distributed anti-malware database management system using blockchain."

[188] S. Rathore, B. W. Kwon, and J. H. Park, "Blockseciotnet: Blockchain-based decentralized security architecture for iot network," *Journal of Network and Computer Applications*, vol. 143, pp. 167–177, 2019.

[189] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini, "Blockchain based decentralized management of demand response programs in smart energy grids," *Sensors*, vol. 18, no. 1, p. 162, 2018.

[190] P. Kochovski, S. Gec, V. Stankovski, M. Bajec, and P. D. Drobintsev, "Trust management in a blockchain based fog computing platform with trustless smart oracles," *Future Generation Computer Systems*, vol. 101, pp. 747–759, 2019.

[191] V. J. Morkunas, J. Paschen, and E. Boon, "How blockchain technologies impact your business model," *Business Horizons*, vol. 62, no. 3, pp. 295–306, 2019.

[192] R. Yang, R. Wakefield, S. Lyu, S. Jayasuriya, F. Han, X. Yi, X. Yang, G. Amarasinghe, and S. Chen, "Public and private blockchain in construction business process and information integration," *Automation in Construction*, vol. 118, p. 103276, 2020.

[193] P. Rimba, A. B. Tran, I. Weber, M. Staples, A. Ponomarev, and X. Xu, "Quantifying the cost of distrust: Comparing blockchain and cloud services for business process execution," *Information Systems Frontiers*, vol. 22, no. 2, pp. 489–507, 2020.

[194] A. Singh, K. Click, R. M. Parizi, Q. Zhang, A. Dehghantanha, and K.-K. R. Choo, "Sidechain technologies in blockchain networks: An examination and state-of-the-art review," *Journal of Network and Computer Applications*, vol. 149, p. 102471, 2020.

[195] P. Zheng, Z. Zheng, X. Luo, X. Chen, and X. Liu, "A detailed and real-time performance monitoring framework for blockchain systems," in *2018 IEEE/ACM 40th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*. IEEE, 2018, pp. 134–143.

[196] S. Smetanin, A. Ometov, M. Komarov, P. Masek, and Y. Koucheryavy, "Blockchain evaluation approaches: State-of-the-art and future perspective," *Sensors*, vol. 20, no. 12, p. 3358, 2020.

[197] H. Sukhwani, N. Wang, K. S. Trivedi, and A. Rindos, "Performance modeling of hyperledger fabric (permissioned blockchain network)," in *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*. IEEE, 2018, pp. 1–8.

[198] T. T. A. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, and K.-L. Tan, "Blockbench: A framework for analyzing private blockchains," in *Proceedings of the 2017 ACM International Conference on Management of Data*, 2017, pp. 1085–1100.

[199] S. Rouhani and R. Deters, "Performance analysis of ethereum transactions in private blockchain," in *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*. IEEE, 2017, pp. 70–74.

[200] P. Thakkar, S. Nathan, and B. Viswanathan, "Performance benchmarking and optimizing hyperledger fabric blockchain platform," in *2018 IEEE 26th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*. IEEE, 2018, pp. 264–276.

[201] A. Baliga, I. Subhod, P. Kamat, and S. Chatterjee, "Performance evaluation of the quorum blockchain platform," *arXiv preprint arXiv:1809.03421*, 2018.

[202] H. Caliper, "Hyperledger caliper architecture," *Electronic Article. url: https://hyperledger. github. io/caliper/docs/2_Architecture. html (visited on 03/10/2019)*, 2019.

[203] B. Ampel, M. Patton, and H. Chen, "Performance modeling of hyperledger sawtooth blockchain," in *2019 IEEE International Conference on Intelligence and Security Informatics (ISI)*. IEEE, 2019, pp. 59–61.

[204] Hyperledger, "hyperledger/iroha." [Online]. Available: https://github.com/hyperledger/iroha

[205] N. L. Hickson-Brown, "Prototyping und evaluierung des hyperledger burrow frameworks unter gesichtspunkten der usability," Ph.D. dissertation, Universität Hamburg, 2019.

[206] Z. Dong, E. Zheng, Y. Choon, and A. Y. Zomaya, "Dagbench: A performance evaluation framework for dag distributed ledgers," in *2019 IEEE 12th International Conference on Cloud Computing (CLOUD)*. IEEE, 2019, pp. 264–271.

[207] M. Divya and N. B. Biradar, "Iota-next generation block chain," *International journal of engineering and computer science*, vol. 7, no. 04, pp. 23 823–23 826, 2018.

[208] A. Churyumov, "Byteball: A decentralized system for storage and transfer of value," *URL https://byteball. org/Byteball. pdf*, 2016.

[209] M. Alharby and A. van Moorsel, "Blocksim: a simulation framework for blockchain systems," *ACM SIGMETRICS Performance Evaluation Review*, vol. 46, no. 3, pp. 135–138, 2019.

[210] S. Pandey, G. Ojha, B. Shrestha, and R. Kumar, "Blocksim: A practical simulation tool for optimal network design, stability and planning." in *2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*. IEEE, 2019, pp. 133–137.

[211] M. Zander, T. Waite, and D. Harz, "Dagsim: Simulation of dag-based distributed ledger protocols," *ACM SIGMETRICS Performance Evaluation Review*, vol. 46, no. 3, pp. 118–121, 2019.

[212] W. F. Silvano and R. Marcelino, "Iota tangle: A cryptocurrency to communicate internet of things data," *Future Generation Computer Systems*, 2020.

[213] M. Kamran, H. U. Khan, W. Nisar, M. Farooq, and S.-U. Rehman, "Blockchain and internet of things: A bibliometric study," *Computers & Electrical Engineering*, vol. 81, p. 106525, 2020.

[214] K. Salah, M. H. U. Rehman, N. Nizamuddin, and A. Al-Fuqaha, "Blockchain for ai: Review and open research challenges," *IEEE Access*, vol. 7, pp. 10 127–10 149, 2019.

[215] H. Jin, Y. Luo, P. Li, and J. Mathew, "A review of secure and privacy-preserving medical data sharing," *IEEE Access*, vol. 7, pp. 61 656–61 669, 2019.

[216] F. B. Kessler, "Basic: Towards a blockchained agent-based simulator for cities," *Massively Multi-Agent Systems II*, p. 144.

[217] J. Constine, "Former employees say lyft staffers spied on passengers," Jan 2018. [Online]. Available: https://techcrunch.com/2018/01/25/lyft-god-view/

[218] L. Fan, J. R. Gil-Garcia, D. Werthmuller, G. B. Burke, and X. Hong, "Investigating blockchain as a data management tool for iot devices in smart city initiatives," in *Proceedings of the 19th Annual International Conference on Digital Government Research: Governance in the Data Age*, 2018, pp. 1–2.

[219] P. Singh, A. Nayyar, A. Kaur, and U. Ghosh, "Blockchain and fog based architecture for internet of everything in smart cities," *Future Internet*, vol. 12, no. 4, p. 61, 2020.

[220] H. Wang, L. Wang, Z. Zhou, X. Tao, G. Pau, and F. Arena, "Blockchain-based resource allocation model in fog computing," *Applied Sciences*, vol. 9, no. 24, p. 5538, 2019.

[221] G. Kumar, R. Saha, M. K. Rai, R. Thomas, and T.-H. Kim, "Proof-of-work consensus approach in blockchain technology for cloud and fog computing using maximization-factorization statistics," *IEEE Internet of Things Journal*, vol. 6, no. 4, pp. 6835–6842, 2019.

[222] S. Schaller and D. Hood, "Software defined networking architecture standardization," *Computer standards & interfaces*, vol. 54, pp. 197–202, 2017.

[223] R. Chaudhary, A. Jindal, G. S. Aujla, S. Aggarwal, N. Kumar, and K.-K. R. Choo, "Best: Blockchain-based secure energy trading in sdn-enabled intelligent transportation system," *Computers & Security*, vol. 85, pp. 288–299, 2019.

[224] P. K. Sharma, S. Singh, Y.-S. Jeong, and J. H. Park, "Distblocknet: A distributed blockchains-based secure sdn architecture for iot networks," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 78–85, 2017.

[225] P. K. Sharma, S. Rathore, Y.-S. Jeong, and J. H. Park, "Softedgenet: Sdn based energy-efficient distributed network architecture for edge computing," *IEEE Communications magazine*, vol. 56, no. 12, pp. 104–111, 2018.

[226] S. Khezr, M. Moniruzzaman, A. Yassine, and R. Benlamri, "Blockchain technology in healthcare: A comprehensive review and directions for future research," *Applied Sciences*, vol. 9, no. 9, p. 1736, 2019.

[227] 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.

[228] H.-T. Pham and P. N. Pathirana, "Measurement and assessment of hand functionality via a cloud-based implementation," in *International Conference on Smart Homes and Health Telematics*. Springer, 2015, pp. 289–294.

[229] S. Li and P. N. Pathirana, "Cloud-based non-invasive tele-rehabilitation exercise monitoring," in *2014 IEEE Conference on Biomedical Engineering and Sciences (IECBES)*. IEEE, 2014, pp. 385–390.

[230] "Hyperledger caliper, retrieved from https://www.hyperledger.org/use/caliper."

[231] A. Celesti, A. Ruggeri, M. Fazio, A. Galletta, M. Villari, and A. Romano, "Blockchain-based healthcare workflow for tele-medical laboratory in federated hospital iot clouds," *Sensors*, vol. 20, no. 9, p. 2590, 2020.

[232] G. Rathee, A. Sharma, H. Saini, R. Kumar, and R. Iqbal, "A hybrid framework for multimedia data processing in iot-healthcare using blockchain technology," *Multimedia Tools and Applications*, pp. 1–23, 2019.

[233] I. Haq and O. M. Esuka, "Blockchain technology in pharmaceutical industry to prevent counterfeit drugs," *Int. J. Comput. Appl.*, vol. 180, no. 25, pp. 8–12, 2018.

[234] D. C. Nguyen, K. D. Nguyen, and P. N. Pathirana, "A mobile cloud based iomt framework for automated health assessment and management," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2019, pp. 6517–6520.

[235] M. Kang, E. Park, B. H. Cho, and K.-S. Lee, "Recent patient health monitoring platforms incorporating internet of things-enabled smart devices," *International neurourology journal*, vol. 22, no. Suppl 2, p. S76, 2018.

[236] S. J. Nass, L. A. Levit, L. O. Gostin *et al.*, "The value and importance of health information privacy," in *Beyond the HIPAA Privacy Rule: Enhancing Privacy, Improving Health Through Research*. National Academies Press (US), 2009.

[237] Y. Rahulamathavan, R. C.-W. Phan, M. Rajarajan, S. Misra, and A. Kondoz, "Privacy-preserving blockchain based iot ecosystem using attribute-based encryption," in *2017 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*. IEEE, 2017, pp. 1–6.

[238] X. Liang, J. Zhao, S. Shetty, J. Liu, and D. Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare applications," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, 2017, pp. 1–5.

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

[240] N. Kahani, K. Elgazzar, and J. R. Cordy, "Authentication and access control in e-health systems in the cloud," in *2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS)*. IEEE, 2016, pp. 13–23.

[241] S. Lu, Y. Hong, Q. Liu, L. Wang, and R. Dssouli, "Access control in e-health portal systems," in *2007 Innovations in Information Technologies (IIT)*. IEEE, 2007, pp. 88–92.

[242] S. R. Islam, M. Hossain, R. Hasan, and T. Q. Duong, "A conceptual framework for an iot-based health assistant and its authorization model," in *2018 IEEE 8th annual computing and communication workshop and conference (CCWC)*. IEEE, 2018, pp. 616–621.

[243] V. Ramani, T. Kumar, A. Bracken, M. Liyanage, and M. Ylianttila, "Secure and efficient data accessibility in blockchain based healthcare systems," in *2018 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2018, pp. 206–212.

[244] H. Wang and Y. Song, "Secure cloud-based ehr system using attribute-based cryptosystem and blockchain," *Journal of medical systems*, vol. 42, no. 8, p. 152, 2018.

[245] T. Hepp, M. Sharinghousen, P. Ehret, A. Schoenhals, and B. Gipp, "On-chain vs. off-chain storage for supply-and blockchain integration," *it-Information Technology*, vol. 60, no. 5-6, pp. 283–291, 2018.

[246] X. Xu, I. Weber, M. Staples, L. Zhu, J. Bosch, L. Bass, C. Pautasso, and P. Rimba, "A taxonomy of blockchain-based systems for architecture design," in *2017 IEEE International Conference on Software Architecture (ICSA)*. IEEE, 2017, pp. 243–252.

[247] X. Liu, Z. Wang, C. Jin, F. Li, and G. Li, "A blockchain-based medical data sharing and protection scheme," *IEEE Access*, vol. 7, pp. 118 943–118 953, 2019.

[248] 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.

[249] Y. Du, J. Liu, Z. Guan, and H. Feng, "A medical information service platform based on distributed cloud and blockchain," in *2018 IEEE International Conference on Smart Cloud (SmartCloud)*. IEEE, 2018, pp. 34–39.

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

[251] K. Fan, S. Wang, Y. Ren, H. Li, and Y. Yang, "Medblock: Efficient and secure medical data sharing via blockchain," *Journal of medical systems*, vol. 42, no. 8, p. 136, 2018.

[252] Z. Kavosi, H. Rahimi, S. Khanian, P. Farhadi, and E. Kharazmi, "Factors influencing decision making for healthcare services outsourcing: A review and delphi study," *Medical journal of the Islamic Republic of Iran*, vol. 32, p. 56, 2018.

[253] H. Skipworth, E. Delbufalo, and C. Mena, "Logistics and procurement outsourcing in the healthcare sector: a comparative analysis," *European Management Journal*, 2020.

[254] H. Zhang, J. Yu, C. Tian, P. Zhao, G. Xu, and J. Lin, "Cloud storage for electronic health records based on secret sharing with verifiable reconstruction outsourcing," *IEEE Access*, vol. 6, pp. 40 713–40 722, 2018.

[255] S. Zhu, Z. Cai, H. Hu, Y. Li, and W. Li, "zkcrowd: a hybrid blockchain-based crowdsourcing platform," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4196–4205, 2019.

[256] X. Xu, Q. Liu, X. Zhang, J. Zhang, L. Qi, and W. Dou, "A blockchain-powered crowdsourcing method with privacy preservation in mobile environment," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1407–1419, 2019.

[257] M. Li, J. Weng, A. Yang, W. Lu, Y. Zhang, L. Hou, J.-N. Liu, Y. Xiang, and R. H. Deng, "Crowdbc: A blockchain-based decentralized framework for crowdsourcing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 6, pp. 1251–1266, 2018.

[258] D. Macrinici, C. Cartofeanu, and S. Gao, "Smart contract applications within blockchain technology: A systematic mapping study," *Telematics and Informatics*, vol. 35, no. 8, pp. 2337–2354, 2018.

[259] Y. Liu, K. Wang, Y. Lin, and W. Xu, "A lightweight blockchain system for industrial internet of things," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 6, pp. 3571–3581, 2019.

[260] L. Ismail, H. Materwala, and S. Zeadally, "Lightweight blockchain for healthcare," *IEEE Access*, vol. 7, pp. 149 935–149 951, 2019.

[261] G. Srivastava, J. Crichigno, and S. Dhar, "A light and secure healthcare blockchain for iot medical devices," in *2019 IEEE Canadian conference of electrical and computer engineering (CCECE)*. IEEE, 2019, pp. 1–5.

[262] O. Attia, I. Khoufi, A. Laouiti, and C. Adjih, "An iot-blockchain architecture based on hyperledger framework for healthcare monitoring application," in *2019 10th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, 2019, pp. 1–5.

[263] J. Yang, M. M. H. Onik, N.-Y. Lee, M. Ahmed, and C.-S. Kim, "Proof-of-familiarity: A privacy-preserved blockchain scheme for collaborative medical decision-making," *Applied Sciences*, vol. 9, no. 7, p. 1370, 2019.

[264] G. Srivastava, A. D. Dwivedi, and R. Singh, "Automated remote patient monitoring: data sharing and privacy using blockchain," *arXiv preprint arXiv:1811.03417*, 2018.

[265] H. Li, H. Tian, F. Zhang, and J. He, "Blockchain-based searchable symmetric encryption scheme," *Computers & Electrical Engineering*, vol. 73, pp. 32–45, 2019.

[266] S. Hu, C. Cai, Q. Wang, C. Wang, X. Luo, and K. Ren, "Searching an encrypted cloud meets blockchain: A decentralized, reliable and fair realization," in *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE, 2018, pp. 792–800.

[267] Y. Wang, A. Zhang, P. Zhang, and H. Wang, "Cloud-assisted ehr sharing with security and privacy preservation via consortium blockchain," *IEEE Access*, vol. 7, pp. 136 704–136 719, 2019.

[268] Y. Xiaodong, L. Ting, L. Rui, and W. Meiding, "Blockchain-based secure and searchable ehr sharing scheme," in *2019 4th International Conference on Mechanical, Control and Computer Engineering (ICMCCE)*. IEEE, 2019, pp. 822–8223.

[269] S.-W. Noh, Y. Park, C. Sur, S.-U. Shin, and K.-H. Rhee, "Blockchain-based user-centric records management system," *Int J Control Autom*, vol. 10, no. 11, pp. 133–144, 2017.

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

[271] N. Islam, Y. Faheem, I. U. Din, M. Talha, M. Guizani, and M. Khalil, "A blockchain-based fog computing framework for activity recognition as an application to e-healthcare services," *Future Generation Computer Systems*, vol. 100, pp. 569–578, 2019.

[272] A. A. Mutlag, M. K. Abd Ghani, N. a. Arunkumar, M. A. Mohammed, and O. Mohd, "Enabling technologies for fog computing in healthcare iot systems," *Future Generation Computer Systems*, vol. 90, pp. 62–78, 2019.

[273] E. Bandara, X. Liang, P. Foytik, S. Shetty, C. Hall, D. Bowden, N. Ranasinghe, and K. De Zoysa, "A blockchain empowered and privacy preserving digital contact tracing platform," *Information Processing & Management*, vol. 58, no. 4, p. 102572, 2021.

[274] D. Pandey, N. Agrawal, and M. P. Jhanwar, "Covidbloc: A blockchain powered exposure database for contact tracing." *IACR Cryptol. ePrint Arch.*, vol. 2020, p. 1543, 2020.

[275] H. Xu, L. Zhang, O. Onireti, Y. Fang, W. J. Buchanan, and M. A. Imran, "Beeptrace: Blockchain-enabled privacy-preserving contact tracing for covid-19 pandemic and beyond," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3915–3929, 2020.

[276] M. Rimsan, A. K. Mahmood, M. Umair, and F. Hassan, "Covid-19: A novel framework to globally track coronavirus infected patients using blockchain," in *2020 International Conference on Computational Intelligence (ICCI)*. IEEE, 2020, pp. 70–74.

[277] A. Azim, M. N. Islam, and P. E. Spranger, "Blockchain and novel coronavirus: Towards preventing covid-19 and future pandemics," *Iberoamerican Journal of Medicine*, vol. 2, no. 3, pp. 215–218, 2020.

[278] S. K. Deka, S. Goswami, and A. Anand, "A blockchain based technique for storing vaccination records," in *2020 IEEE Bombay Section Signature Conference (IBSSC)*. IEEE, 2020, pp. 135–139.

[279] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Privacy-preserved task offloading in mobile blockchain with deep reinforcement learning," *arXiv preprint arXiv:1908.07467*, 2019.

[280] L. Hang, I. Ullah, and D.-H. Kim, "A secure fish farm platform based on blockchain for agriculture data integrity," *Computers and Electronics in Agriculture*, vol. 170, p. 105251, 2020.

[281] Y. Zhang, R. H. Deng, X. Liu, and D. Zheng, "Blockchain based efficient and robust fair payment for outsourcing services in cloud computing," *Information Sciences*, vol. 462, pp. 262–277, 2018.

[282] B. Alessio, W. De Donato, V. Persico, and A. Pescapé, "On the integration of cloud computing and internet of things," in *Future Internet of Things and Cloud (FiCloud), 2014 International Conference on. IEEE*, 2014.

[283] B. L. R. Stojkoska and K. V. Trivodaliev, "A review of internet of things for smart home: Challenges and solutions," *Journal of Cleaner Production*, vol. 140, pp. 1454–1464, 2017.

[284] M. Sookhak, H. Tang, Y. He, and F. R. Yu, "Security and privacy of smart cities: a survey, research issues and challenges," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1718–1743, 2018.

[285] R. Paul, P. Baidya, S. Sau, K. Maity, S. Maity, and S. B. Mandal, "Iot based secure smart city architecture using blockchain," in *2018 2nd International Conference on Data Science and Business Analytics (ICDSBA)*. IEEE, 2018, pp. 215–220.

[286] M. AbuNaser and A. A. Alkhatib, "Advanced survey of blockchain for the internet of things smart home," in *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*. IEEE, 2019, pp. 58–62.

[287] A. Dorri, S. S. Kanhere, and R. Jurdak, "Towards an optimized blockchain for iot." ACM, 2017, pp. 173–178.

[288] S. Singh, I.-H. Ra, W. Meng, M. Kaur, and G. H. Cho, "Sh-blockcc: A secure and efficient internet of things smart home architecture based on cloud computing and blockchain technology," *International Journal of Distributed Sensor Networks*, vol. 15, no. 4, p. 1550147719844159, 2019.

[289] J. Xue, C. Xu, and Y. Zhang, "Private blockchain-based secure access control for smart home systems." *KSII Transactions on Internet & Information Systems*, vol. 12, no. 12, 2018.

[290] P. K. Singh, R. Singh, S. K. Nandi, and S. Nandi, "Managing smart home appliances with proof of authority and blockchain," in *International Conference on Innovations for Community Services*. Springer, 2019, pp. 221–232.

[291] J. Ali, A. S. Khalid, E. Yafi, S. Musa, and W. Ahmed, "Towards a secure behavior modeling for iot networks using blockchain," *arXiv preprint arXiv:2001.01841*, 2020.

[292] B. Yin, L. Mei, Z. Jiang, and K. Wang, "Joint cloud collaboration mechanism between vehicle clouds based on blockchain," in *2019 IEEE International Conference on Service-Oriented System Engineering (SOSE)*. IEEE, 2019, pp. 227–2275.

[293] H. Liu, Y. Zhang, and T. Yang, "Blockchain-enabled security in electric vehicles cloud and edge computing," *IEEE Network*, vol. 32, no. 3, pp. 78–83, 2018.

[294] S. Nadeem, M. Rizwan, F. Ahmad, and J. Manzoor, "Securing cognitive radio vehicular ad hoc network with fog node based distributed blockchain cloud architecture," *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, vol. 10, no. 1, pp. 288–295, 2019.

[295] L. Xie, Y. Ding, H. Yang, and X. Wang, "Blockchain-based secure and trustworthy internet of things in sdn-enabled 5g-vanets," *IEEE Access*, vol. 7, pp. 56 656–56 666, 2019.

[296] A. R. Pedrosa and G. Pau, "Chargeltup: On blockchain-based technologies for autonomous vehicles," in *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, 2018, pp. 87–92.

[297] M. Li, L. Zhu, and X. Lin, "Efficient and privacy-preserving carpooling using blockchain-assisted vehicular fog computing," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4573–4584, 2018.

[298] Y. Yao, X. Chang, J. Mišić, V. B. Mišić, and L. Li, "Bla: Blockchain-assisted lightweight anonymous authentication for distributed vehicular fog services," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3775–3784, 2019.

[299] J. Gao, K. O.-B. O. Agyekum, E. B. Sifah, K. N. Acheampong, Q. Xia, X. Du, M. Guizani, and H. Xia, "A blockchain-sdn-enabled internet of vehicles environment for fog computing and 5g networks," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4278–4291, 2019.

[300] V. Hassija, V. Saxena, V. Chamola, and R. Yu, "A parking slot allocation framework based on virtual voting and adaptive pricing algorithm," *IEEE Transactions on Vehicular Technology*, 2020.

[301] V. Hassija, V. Chamola, S. Garg, N. G. K. Dara, G. Kaddoum, and D. N. K. Jayakody, "A blockchain-based framework for lightweight data sharing and energy trading in v2g network," *IEEE Transactions on Vehicular Technology*, 2020.

[302] V. Hassija, V. Gupta, S. Garg, and V. Chamola, "Traffic jam probability estimation based on blockchain and deep neural networks," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[303] S. Biswas, K. Sharif, F. Li, B. Nour, and Y. Wang, "A scalable blockchain framework for secure transactions in iot," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4650–4659, 2018.

[304] M. Pourvahab and G. Ekbatanifard, "An efficient forensics architecture in software-defined networking-iot using blockchain technology," *IEEE Access*, vol. 7, pp. 99 573–99 588, 2019.

[305] A. S. Hosen, S. Singh, P. K. Sharma, U. Ghosh, J. Wang, I.-H. Ra, and G. H. Cho, "Blockchain-based transaction validation protocol for a secure distributed iot network," *IEEE Access*, 2020.

[306] A. Muthanna, A. A. Ateya, A. Khakimov, I. Gudkova, A. Abuarqoub, K. Samouylov, and A. Koucheryavy, "Secure iot network structure based on distributed fog computing, with sdn/blockchain," 2019.

[307] Y. Gao, Y. Chen, H. Lin, and J. J. Rodrigues, "Blockchain based secure iot data sharing framework for sdn-enabled smart communities," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2020, pp. 514–519.

[308] S. Misra, P. K. Deb, N. Pathak, and A. Mukherjee, "Blockchain-enabled sdn for securing fog-based resource-constrained iot," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2020, pp. 490–495.

[309] P. Zhang, F. Liu, N. Kumar, and G. S. Aujla, "Information classification strategy for blockchain-based secure sdn in iot scenario," in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 2020, pp. 1081–1086.

[310] P. T. Duy, H. Do Hoang, N. B. Khanh, V.-H. Pham *et al.*, "Sdnlog-foren: Ensuring the integrity and tamper resistance of log files for sdn forensics using blockchain," in *2019 6th NAFOSTED Conference on Information and Computer Science (NICS)*. IEEE, 2019, pp. 416–421.

[311] D. V. Medhane, A. K. Sangaiah, M. S. Hossain, G. Muhammad, and J. Wang, "Blockchain-enabled distributed security framework for next generation iot: An edge-cloud and software defined network integrated approach," *IEEE Internet of Things Journal*, 2020.

[312] Z. Abou El Houda, A. Hafid, and L. Khoukhi, "Co-iot: A collaborative ddos mitigation scheme in iot environment based on blockchain using sdn," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[313] V. Hassija, V. Saxena, and V. Chamola, "A mobile data offloading framework based on a combination of blockchain and virtual voting," *Software: Practice and Experience*, 2020.

[314] Z. Xiong, Y. Zhang, D. Niyato, P. Wang, and Z. Han, "When mobile blockchain meets edge computing," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 33–39, 2018.

[315] Y. Jiao, P. Wang, D. Niyato, and K. Suankaewmanee, "Auction mechanisms in cloud/fog computing resource allocation for public blockchain networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 9, pp. 1975–1989, 2019.

[316] W. Tang, X. Zhao, W. Rafique, and W. Dou, "A blockchain-based offloading approach in fog computing environment," in *2018 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications (ISPA/IUCC/BDCloud/SocialCom/SustainCom)*. IEEE, 2018, pp. 308–315.

[317] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "Privacy-preserved task offloading in mobile blockchain with deep reinforcement learning," *IEEE Transactions on Network and Service Management*, 2020.

[318] Z. Noshad, A. Javaid, M. Zahid, I. Ali, N. Javaid *et al.*, "Node recovery in wireless sensor networks via blockchain," in *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*. Springer, 2019, pp. 94–105.

[319] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasurbramanian, "A lightweight blockchain based framework for underwater iot," *Electronics*, vol. 8, no. 12, p. 1552, 2019.

[320] H. L. Cech, M. Großmann, and U. R. Krieger, "A fog computing architecture to share sensor data by means of blockchain functionality," in *2019 IEEE International Conference on Fog Computing (ICFC)*. IEEE, 2019, pp. 31–40.

[321] X. Zhu and Y. Badr, "Fog computing security architecture for the internet of things using blockchain-based social networks," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. IEEE, 2018, pp. 1361–1366.

[322] O. Samuel, N. Javaid, M. Awais, Z. Ahmed, M. Imran, and M. Guizani, "A blockchain model for fair data sharing in deregulated smart grids," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–7.

[323] V. Hassija, V. Chamola, V. Gupta, S. Jain, and N. Guizani, "A survey on supply chain security: Application areas, security threats, and solution architectures," *IEEE Internet of Things Journal*, 2020.

[324] S. Malik, S. S. Kanhere, and R. Jurdak, "Productchain: Scalable blockchain framework to support provenance in supply chains," in *2018 IEEE 17th International Symposium on Network Computing and Applications (NCA)*. IEEE, 2018, pp. 1–10.

[325] S. Figorilli, F. Antonucci, C. Costa, F. Pallottino, L. Raso, M. Castiglione, E. Pinci, D. Del Vecchio, G. Colle, A. R. Proto *et al.*, "A blockchain implementation prototype for the electronic open source traceability of wood along the whole supply chain," *Sensors*, vol. 18, no. 9, p. 3133, 2018.

[326] R. Martínez-Peláez, H. Toral-Cruz, J. R. Parra-Michel, V. García, L. J. Mena, V. G. Félix, and A. Ochoa-Brust, "An enhanced lightweight iot-based authentication scheme in cloud computing circumstances," *Sensors*, vol. 19, no. 9, p. 2098, 2019.

[327] L. Zhou, X. Li, K.-H. Yeh, C. Su, and W. Chiu, "Lightweight iot-based authentication scheme in cloud computing circumstance," *Future Generation Computer Systems*, vol. 91, pp. 244–251, 2019.

[328] M. Ma, G. Shi, and F. Li, "Privacy-oriented blockchain-based distributed key management architecture for hierarchical access control in the iot scenario," *IEEE Access*, vol. 7, pp. 34 045–34 059, 2019.

[329] Z. Bao, W. Shi, D. He, and K.-K. R. Chood, "Iotchain: A three-tier blockchain-based iot security architecture," *arXiv preprint arXiv:1806.02008*, 2018.

[330] A. Seitz, D. Henze, D. Miehle, B. Bruegge, J. Nickles, and M. Sauer, "Fog computing as enabler for blockchain-based iiot app marketplaces-a case study," in *2018 Fifth international conference on internet of things: systems, management and security*. IEEE, 2018, pp. 182–188.

[331] M. Babaioff, N. Immorlica, D. Kempe, and R. Kleinberg, "Online auctions and generalized secretary problems," *ACM SIGecom Exchanges*, vol. 7, no. 2, pp. 1–11, 2008.

[332] N. Bildirici, "System and method for comprehensive remote patient monitoring and management," Feb. 8 2007, uS Patent App. 11/198,586.

[333] M. Chan, D. Estève, J.-Y. Fourniols, C. Escriba, and E. Campo, "Smart wearable systems: Current status and future challenges," *Artificial intelligence in medicine*, vol. 56, no. 3, pp. 137–156, 2012.

[334] S. Alasmari and M. Anwar, "Security & privacy challenges in iot-based health cloud," in *Computational Science and Computational Intelligence (CSCI), 2016 International Conference on*. IEEE, 2016, pp. 198–201.

[335] R. Clarke, "Introduction to dataveillance and information privacy, and definitions of terms," *Roger Clarke's Dataveillance and Information Privacy Pages*, 1999, Available from http://www.rogerclarke.com/DV/Privacy.html, Accessed 10 October, 2017.

[336] P. PWC, "Managing cyber risks in an interconnected world: Key findings from the global state of information security survey 2015," 2015.

[337] L. Malina, J. Hajny, R. Fujdiak, and J. Hosek, "On perspective of security and privacy-preserving solutions in the internet of things," *Computer Networks*, vol. 102, pp. 83–95, 2016.

[338] ISO, "27799-health informatics-information security management in health using iso," *IEC*, 2008.

[339] P. Gope and T. Hwang, "Bsn-care: A secure iot-based modern healthcare system using body sensor network," *IEEE Sensors Journal*, vol. 16, no. 5, pp. 1368–1376, 2016.

[340] H. Wang, D. Peng, W. Wang, H. Sharif, H.-H. Chen, and A. Khoynezhad, "Resource-aware secure ecg healthcare monitoring through body sensor networks," *IEEE Wireless Communications*, vol. 17, no. 1, 2010.

[341] E. Al Alkeem, D. Shehada, C. Y. Yeun, M. J. Zemerly, and J. Hu, "New secure healthcare system using cloud of things," *Cluster Computing*, vol. 20, no. 3, pp. 2211–2229, 2017.

[342] M. C. Chuah and F. Fu, "Ecg anomaly detection via time series analysis," in *International Symposium on Parallel and Distributed Processing and Applications*. Springer, 2007, pp. 123–135.

[343] H. Sivaraks and C. A. Ratanamahatana, "Robust and accurate anomaly detection in ecg artifacts using time series motif discovery," *Computational and mathematical methods in medicine*, vol. 2015, 2015.

[344] P. Ghorbanian, A. Ghaffari, A. Jalali, and C. Nataraj, "Heart arrhythmia detection using continuous wavelet transform and principal component analysis with neural network classifier," in *Computing in Cardiology, 2010*. IEEE, 2010, pp. 669–672.

[345] L. Clifton, D. A. Clifton, M. A. Pimentel, P. J. Watkinson, and L. Tarassenko, "Predictive monitoring of mobile patients by combining clinical observations with data from wearable sensors," *IEEE journal of biomedical and health informatics*, vol. 18, no. 3, pp. 722–730, 2014.

[346] C. Zenger, "Physical-layer security for the internet of things," *(Doctor-Engineer), University of Bochum, Germany*, 2017.

[347] N. Ye, Y. Zhu, R.-c. Wang, R. Malekian, and L. Qiao-min, "An efficient authentication and access control scheme for perception layer of internet of things," *Applied Mathematics & Information Sciences*, vol. 8, no. 4, p. 1617, 2014.

[348] D. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton, "Codeblue: An ad hoc sensor network infrastructure for emergency medical care," in *International workshop on wearable and implantable body sensor networks*, vol. 5. Boston, MA;, 2004.

[349] A. Wood, G. Virone, T. Doan, Q. Cao, L. Selavo, Y. Wu, L. Fang, Z. He, S. Lin, and J. Stankovic, "Alarm-net: Wireless sensor networks for assisted-living and residential monitoring," *University of Virginia Computer Science Department Technical Report*, vol. 2, p. 17, 2006.

[350] S. Pai, M. Meingast, T. Roosta, S. Bermudez, S. B. Wicker, D. K. Mulligan, and S. Sastry, "Transactional confidentiality in sensor networks," *IEEE Security & Privacy*, vol. 6, no. 4, 2008.

[351] P. Kumar and H.-J. Lee, "Security issues in healthcare applications using wireless medical sensor networks: A survey," *Sensors*, vol. 12, no. 1, pp. 55–91, 2011.

[352] J. W. Ng, B. P. Lo, O. Wells, M. Sloman, N. Peters, A. Darzi, C. Toumazou, and G.-Z. Yang, "Ubiquitous monitoring environment for wearable and implantable sensors (ubimon)," in *International Conference on Ubiquitous Computing (Ubicomp)*, 2004.

[353] R. Chakravorty, "A programmable service architecture for mobile medical care," in *Pervasive Computing and Communications Workshops, 2006. PerCom Workshops 2006. Fourth Annual IEEE International Conference on*. IEEE, 2006, pp. 5–pp.

[354] K. Malasri and L. Wang, "Snap: an architecture for secure medical sensor networks," in *Wireless Mesh Networks, 2006. WiMesh 2006. 2nd IEEE Workshop on*. IEEE, 2006, pp. 160–162.

[355] J. Ko, J. H. Lim, Y. Chen, R. Musvaloiu-E, A. Terzis, G. M. Masson, T. Gao, W. Destler, L. Selavo, and R. P. Dutton, "Medisn: Medical emergency detection in sensor networks," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 10, no. 1, p. 11, 2010.

[356] S. R. Moosavi, T. N. Gia, E. Nigussie, A. M. Rahmani, S. Virtanen, H. Tenhunen, and J. Isoaho, "End-to-end security scheme for mobility enabled healthcare internet of things," *Future Generation Computer Systems*, vol. 64, pp. 108–124, 2016.

[357] K.-H. Yeh, "A secure iot-based healthcare system with body sensor networks," *IEEE Access*, vol. 4, pp. 10 288–10 299, 2016.

[358] H. Krawczyk, R. Canetti, and M. Bellare, "Hmac: Keyed-hashing for message authentication," February 1997.

[359] H. Yang and V. A. Oleshchuk, "A dynamic attribute-based authentication scheme," in *International Conference on Codes, Cryptology, and Information Security*. Springer, 2015, pp. 106–118.

[360] X. Liang, R. Lu, L. Chen, X. Lin, and X. Shen, "Pec: A privacy-preserving emergency call scheme for mobile healthcare social networks," *Journal of Communications and Networks*, vol. 13, no. 2, pp. 102–112, 2011.

[361] A. Lounis, A. Hadjidj, A. Bouabdallah, and Y. Challal, "Secure and scalable cloud-based architecture for e-health wireless sensor networks," in *Computer communications and networks (ICCCN), 2012 21st international conference on*. IEEE, 2012, pp. 1–7.

[362] X. Liang, X. Li, R. Lu, X. Lin, and X. Shen, "Enabling pervasive healthcare with privacy preservation in smart community," in *Communications (ICC), 2012 IEEE International Conference on*. IEEE, 2012, pp. 3451–3455.

[363] M. Li, S. Yu, Y. Zheng, K. Ren, and W. Lou, "Scalable and secure sharing of personal health records in cloud computing using attribute-based encryption," *IEEE transactions on parallel and distributed systems*, vol. 24, no. 1, pp. 131–143, 2013.

[364] S. B. Baker, W. Xiang, and I. Atkinson, "Internet of things for smart healthcare: Technologies, challenges, and opportunities," *IEEE Access*, vol. 5, pp. 26 521–26 544, 2017.

[365] C. Stagnaro, "White paper: Innovative blockchain uses in health care," *Available from www.freedassociates.com, Accessed 1 April, 2018*.

[366] A. Dorri, S. S. Kanhere, and R. Jurdak, "Towards an optimized blockchain for iot." ACM, 2017, pp. 173–178.

[367] T.-T. Kuo, H.-E. Kim, and L. Ohno-Machado, "Blockchain distributed ledger technologies for biomedical and health care applications," *Journal of the American Medical Informatics Association*, vol. 24, no. 6, pp. 1211–1220, 2017.

[368] M. Haghi, K. Thurow, and R. Stoll, "Wearable devices in medical internet of things: Scientific research and commercially available devices," *Healthcare informatics research*, vol. 23, no. 1, pp. 4–15, 2017.

[369] D. Wu, B. Yang, H. Wang, D. Wu, and R. Wang, "An energy-efficient data forwarding strategy for heterogeneous wbans," *IEEE Access*, vol. 4, pp. 7251–7261, 2016.

[370] S. J. Preece, J. Y. Goulermas, L. P. Kenney, D. Howard, K. Meijer, and R. Crompton, "Activity identification using body-mounted sensors—a review of classification techniques," *Physiological measurement*, vol. 30, no. 4, p. R1, 2009.

[371] J. Shen, S. Chang, J. Shen, Q. Liu, and X. Sun, "A lightweight multi-layer authentication protocol for wireless body area networks," *Future Generation Computer Systems*, vol. 78, pp. 956–963, 2018.

[372] N. Z. Gong, A. Ozen, Y. Wu, X. Cao, R. Shin, D. Song, H. Jin, and X. Bao, "Piano: Proximity-based user authentication on voice-powered internet-of-things devices," *arXiv preprint arXiv:1704.03118*, 2017.

[373] H. Shafagh and A. Hithnawi, "Poster: come closer: proximity-based authentication for the internet of things." Proceedings of the 20th annual international conference on Mobile computing and networking,ACM, 2014, pp. 421–424.

[374] J. Zhang, Z. Wang, Z. Yang, and Q. Zhang, "Proximity based iot device authentication," *In INFOCOM 2017-IEEE Conference on Computer Communications*, pp. pp. 1–9, 2017.

[375] G. C. Pereira, R. C. Alves, F. L. d. Silva, R. M. Azevedo, B. C. Albertini, and C. B. Margi, "Performance evaluation of cryptographic algorithms over iot platforms and operating systems," *Security and Communication Networks*, vol. 2017, 2017.

[376] J. Yin, Q. Yang, and J. J. Pan, "Sensor-based abnormal human-activity detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 8, pp. 1082–1090, 2008.

[377] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors." *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013.

[378] A. Darwish and A. E. Hassanien, "Wearable and implantable wireless sensor network solutions for healthcare monitoring," *Sensors*, vol. 11, no. 6, pp. 5561–5595, 2011.

[379] H. Mamaghanian, N. Khaled, D. Atienza, and P. Vandergheynst, "Compressed sensing for real-time energy-efficient ecg compression on wireless body sensor nodes," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 9, pp. 2456–2466, 2011.

[380] S. Li, L. Da Xu, and X. Wang, "A continuous biomedical signal acquisition system based on compressed sensing in body sensor networks," *IEEE transactions on industrial informatics*, vol. 9, no. 3, pp. 1764–1771, 2013.

[381] A. M. Dixon, E. G. Allstot, D. Gangopadhyay, and D. J. Allstot, "Compressed sensing system considerations for ecg and emg wireless biosensors," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 6, no. 2, pp. 156–166, 2012.

[382] T. Eisenbarth and S. Kumar, "A survey of lightweight-cryptography implementations," *IEEE Design & Test of Computers*, vol. 24, no. 6, 2007.

[383] T. Li, J. Ren, and X. Tang, "Secure wireless monitoring and control systems for smart grid and smart home," *IEEE Wireless Communications*, vol. 19, no. 3, 2012.

[384] K. Theocharoulis, I. Papaefstathiou, and C. Manifavas, "Implementing rainbow tables in high-end fpgas for super-fast password cracking," in *Field Programmable Logic and Applications (FPL), 2010 International Conference on*. IEEE, 2010, pp. 145–150.

[385] H. H. Ngo, X. Wu, P. D. Le, C. Wilson, and B. Srinivasan, "Dynamic key cryptography and applications." *IJ Network Security*, vol. 10, no. 3, pp. 161–174, 2010.

[386] U. Mukhopadhyay, A. Skjellum, O. Hambolu, J. Oakley, L. Yu, and R. Brooks, "A brief survey of cryptocurrency systems," in *Privacy, Security and Trust (PST), 2016 14th Annual Conference on*. IEEE, 2016, pp. 745–752.

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

[388] S. Park, K. Pietrzak, J. Alwen, G. Fuchsbauer, and P. Gazi, "Spacecoin: A cryptocurrency based on proofs of space," IACR Cryptology ePrint Archive 2015, Tech. Rep., 2015.

[389] N. Z. Aitzhan and D. Svetinovic, "Security and privacy in decentralized energy trading through multi-signatures, blockchain and anonymous messaging streams," *IEEE Transactions on Dependable and Secure Computing*, 2016.

[390] O. Informatics, "Clinical knowledge manager," *OpenEHR Clinical Knowledge Manager*, 2015.

[391] R. Ramachandran, S. Neelakantan, and A. S. Bidyarthy, "Behavior model for detecting data exfiltration in network environment," in *Internet Multimedia Systems Architecture and Application (IMSAA), 2011 IEEE 5th International Conference on*. IEEE, 2011, pp. 1–5.

[392] Y. Liu, C. Corbett, K. Chiang, R. Archibald, B. Mukherjee, and D. Ghosal, "Sidd: A framework for detecting sensitive data exfiltration by an insider attack," in *System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference on*. IEEE, 2009, pp. 1–10.

[393] V. Balasubramanian, D. B. Hoang, and T. A. Zia, "Addressing the confidentiality and integrity of assistive care loop framework using wireless sensor networks," in *Systems Engineering (ICSEng), 2011 21st International Conference on*. IEEE, 2011, pp. 416–421.

[394] A. Dresch, D. P. Lacerda, and J. A. V. Antunes, "Design science research," in *Design science research*. Springer, 2015, pp. 67–102.

[395] I. Koren and C. M. Krishna, *Fault-tolerant systems*. Morgan Kaufmann, 2020.

[396] C. Systems, "Fog computing and the internet of things: Extend the cloud to where the things are," 2015.

[397] L. Goubin and A. Martinelli, "Protecting aes with shamir's secret sharing scheme," in *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 2011, pp. 79–94.

[398] S. Noether, "Ring signature confidential transactions for monero." *IACR Cryptol. ePrint Arch.*, vol. 2015, p. 1098, 2015.

[399] R. Abdolkhani, K. Gray, A. Borda, and R. DeSouza, "Patient-generated health data management and quality challenges in remote patient monitoring," *JAMIA Open*, 2019.

[400] M. Chernyshev, S. Zeadally, and Z. Baig, "Healthcare data breaches: Implications for digital forensic readiness," *Journal of medical systems*, vol. 43, no. 1, p. 7, 2019.

[401] J. Zou *et al.*, "Accountability in cloud services," Ph.D. dissertation, Macquarie University, Faculty of Science and Engineering, Department of . . . , 2016.

[402] S. Safavat, N. N. S. Naveen, and D. B. Rawat, "Recent advances in mobile edge computing and content caching," *Digital Communications and Networks*, 2019.

[403] R. Gibbings and N. Wickramasinghe, "A systematic framework to assess emrs and ehrs," in *Theories to Inform Superior Health Informatics Research and Practice*. Springer, 2018, pp. 403–413.

[404] T. Heart, O. Ben-Assuli, and I. Shabtai, "A review of phr, emr and ehr integration: A more personalized healthcare and public health policy," *Health Policy and Technology*, vol. 6, no. 1, pp. 20–25, 2017.

[405] G. Drosatos and E. Kaldoudi, "Blockchain applications in the biomedical domain: A scoping review," *Computational and Structural Biotechnology Journal*, 2019.

[406] P. Dunphy and F. A. Petitcolas, "A first look at identity management schemes on the blockchain," *IEEE Security & Privacy*, vol. 16, no. 4, pp. 20–29, 2018.

[407] 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, 2016, p. 13.

[408] 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.

[409] A. Lei, H. Cruickshank, Y. Cao, P. Asuquo, C. P. A. Ogah, and Z. Sun, "Blockchain-based dynamic key management for heterogeneous intelligent transportation systems," *IEEE Internet of Things Journal*, 2017.

[410] V. N. Inukollu, T. Kang, and N. Sakhnini, "Design constraints and challenges behind fault tolerance systems in a mobile application framework," in *2015 10th International Design & Test Symposium (IDT)*. IEEE, 2015, pp. 159–160.

[411] N. Alliance, "Description of network slicing concept," *NGMN 5G P*, vol. 1, 2016.

[412] J. Ordonez-Lucena, P. Ameigeiras, D. Lopez, J. J. Ramos-Munoz, J. Lorca, and J. Folgueira, "Network slicing for 5g with sdn/nfv: Concepts, architectures, and challenges," *IEEE Communications Magazine*, vol. 55, no. 5, pp. 80–87, 2017.

[413] M. Chen, Y. Qian, Y. Hao, Y. Li, and J. Song, "Data-driven computing and caching in 5g networks: Architecture and delay analysis," *IEEE Wireless Communications*, vol. 25, no. 1, pp. 70–75, 2018.

[414] P. Verma and S. K. Sood, "Cloud-centric iot based disease diagnosis healthcare framework," *Journal of Parallel and Distributed Computing*, vol. 116, pp. 27–38, 2018.

[415] C. O. Fernandes and C. J. P. De Lucena, "A software framework for remote patient monitoring by using multi-agent systems support," *JMIR medical informatics*, vol. 5, no. 1, 2017.

[416] D. Yuan, J. Jin, J. Grundy, and Y. Yang, "A framework for convergence of cloud services and internet of things," in *Computer Supported Cooperative Work in Design (CSCWD), 2015 IEEE 19th International Conference on*. IEEE, 2015, pp. 349–354.

[417] M. A. Rahman, M. S. Hossain, G. Loukas, E. Hassanain, S. S. Rahman, M. F. Alhamid, and M. Guizani, "Blockchain-based mobile edge computing framework for secure therapy applications," *IEEE Access*, vol. 6, pp. 72 469–72 478, 2018.

[418] X. Liang, J. Zhao, S. Shetty, J. Liu, and D. Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare applications," in *2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Oct 2017, pp. 1–5.

[419] J. Brogan, I. Baskaran, and N. Ramachandran, "Authenticating health activity data using distributed ledger technologies," *Computational and Structural Biotechnology Journal*, vol. 16, pp. 257–266, 2018.

[420] W. J. Gordon and C. Catalini, "Blockchain technology for healthcare: facilitating the transition to patient-driven interoperability," *Computational and structural biotechnology journal*, vol. 16, pp. 224–230, 2018.

[421] T. Rupasinghe, F. Burstein, C. Rudolph, and S. Strange, "Towards a blockchain based fall prediction model for aged care," in *Proceedings of the Australasian Computer Science Week Multiconference*. ACM, 2019, p. 32.

[422] E. Gaetani, L. Aniello, R. Baldoni, F. Lombardi, A. Margheri, and V. Sassone, "Blockchain-based database to ensure data integrity in cloud computing environments," 2017.

[423] O. Novo, "Blockchain meets iot: An architecture for scalable access management in iot," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1184–1195, 2018.

[424] R. Guo, H. Shi, Q. Zhao, and D. Zheng, "Secure attribute-based signature scheme with multiple authorities for blockchain in electronic health records systems," *IEEE Access*, vol. 6, pp. 11 676–11 686, 2018.

[425] S. Yaqoob, M. M. Khan, R. Talib, A. D. Butt, S. Saleem, F. Arif, and A. Nadeem, "Use of blockchain in healthcare: A systematic literature review," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 5, 2019.

[426] S. P. Amaraweera and M. N. Halgamuge, "Internet of things in the healthcare sector: Overview of security and privacy issues," in *Security, Privacy and Trust in the IoT Environment*. Springer, 2019, pp. 153–179.

[427] L. Faramondi, G. Oliva, R. Setola, and L. Vollero, "Iiot in the hospital scenario: Hospital 4.0, blockchain and robust data management," in *Security and Privacy Trends in the Industrial Internet of Things*. Springer, 2019, pp. 271–285.

[428] M. Chen, Y. Zhang, Y. Li, S. Mao, and V. C. Leung, "Emc: Emotion-aware mobile cloud computing in 5g," *IEEE Network*, vol. 29, no. 2, pp. 32–38, 2015.

[429] K. Lin, F. Xia, W. Wang, D. Tian, and J. Song, "System design for big data application in emotion-aware healthcare," *IEEE Access*, vol. 4, pp. 6901–6909, 2016.

[430] M. S. Hossain and G. Muhammad, "Emotion-aware connected healthcare big data towards 5g," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 2399–2406, 2017.

[431] M. Chen, J. Yang, J. Zhou, Y. Hao, J. Zhang, and C.-H. Youn, "5g-smart diabetes: Toward personalized diabetes diagnosis with healthcare big data clouds," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 16–23, 2018.

[432] F. Gai, B. Wang, W. Deng, and W. Peng, "Proof of reputation: a reputation-based consensus protocol for peer-to-peer network," in *International Conference on Database Systems for Advanced Applications*. Springer, 2018, pp. 666–681.

[433] K. Li, H. Li, H. Hou, K. Li, and Y. Chen, "Proof of vote: A high-performance consensus protocol based on vote mechanism & consortium blockchain," in *2017 IEEE 19th International Conference on High Performance Computing and Communications; IEEE 15th International Conference on Smart City; IEEE 3rd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*. IEEE, 2017, pp. 466–473.

[434] I. Eyal, A. E. Gencer, E. G. Sirer, and R. Van Renesse, "Bitcoin-ng: A scalable blockchain protocol." in *NSDI*, 2016, pp. 45–59.

[435] K. Peterson, R. Deeduvanu, P. Kanjamala, and K. Boles, "A blockchain-based approach to health information exchange networks," in *Proc. NIST Workshop Blockchain Healthcare*, vol. 1, 2016, pp. 1–10.

[436] G. Greenspan, "Multichain private blockchain—white paper," *.[Online]. Available: http://www. multichain. com/download/MultiChain-White-Paper. pdf. [Accessed: 19-Sept-2018].*, 2015.

[437] Y. Lu, "Blockchain: A survey on functions, applications and open issues," *Journal of Industrial Integration and Management*, vol. 3, no. 04, p. 1850015, 2018.

[438] A. Cuervo, D.-k. Cho, A. Wolman, S. Saroiu, R. Chandra, and P. Bahl, "Making smartphones last longer with code offload," in *8th international conference on Mobile systems, applications, and services*, pp. 49–62.

[439] B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "Clonecloud: elastic execution between mobile device and cloud," in *Proceedings of the sixth conference on Computer systems*. ACM, 2011, pp. 301–314.

[440] S. Kosta, A. Aucinas, P. Hui, R. Mortier, and X. Zhang, "Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading," in *2012 Proceedings IEEE Infocom*. IEEE, 2012, pp. 945–953.

[441] T.-Y. Lin, T.-A. Lin, C.-H. Hsu, and C.-T. King, "Context-aware decision engine for mobile cloud offloading," in *2013 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. IEEE, 2013, pp. 111–116.

[442] M. E. Khoda, M. A. Razzaque, A. Almogren, M. M. Hassan, A. Alamri, and A. Alelaiwi, "Efficient computation offloading decision in mobile cloud computing over 5g network," *Mobile Networks and Applications*, vol. 21, no. 5, pp. 777–792, 2016.

[443] H. Yang and J. M. Garibaldi, "Automatic detection of protected health information from clinic narratives," *Journal of biomedical informatics*, vol. 58, pp. S30–S38, 2015.

[444] P. Jindal, D. Roth, and C. A. Gunter, "Detecting privacy-sensitive events in medical text," Tech. Rep., 2013.

[445] D. Sánchez, M. Batet, and A. Viejo, "Detecting sensitive information from textual documents: an information-theoretic approach," in *International Conference on Modeling Decisions for Artificial Intelligence*. Springer, 2012, pp. 173–184.

[446] W. B. Tesfay and J. Serna-Olvera, "Towards user-centered privacy risk detection and quantification framework," in *2016 8th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, 2016, pp. 1–5.

[447] Y. Zhang, S. Kasahara, Y. Shen, X. Jiang, and J. Wan, "Smart contract-based access control for the internet of things," *arXiv preprint arXiv:1802.04410*, 2018.

[448] I. M. Coelho, V. N. Coelho, R. P. Araujo, W. Yong Qiang, and B. D. Rhodes, "Challenges of pbft-inspired consensus for blockchain and enhancements over neo dbft," *Future Internet*, vol. 12, no. 8, p. 129, 2020.

[449] X. Hao, L. Yu, L. Zhiqiang, L. Zhen, and G. Dawu, "Dynamic practical byzantine fault tolerance," in *2018 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2018, pp. 1–8.

[450] Y. Jiang and Z. Lian, "High performance and scalable byzantine fault tolerance," in *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*. IEEE, 2019, pp. 1195–1202.

[451] S. Lee and S. Kim, "Short selling attack: A self-destructive but profitable 51% attack on pos blockchains." *IACR Cryptol. ePrint Arch.*, vol. 2020, p. 19, 2020.

[452] J. Brown-Cohen, A. Narayanan, A. Psomas, and S. M. Weinberg, "Formal barriers to longest-chain proof-of-stake protocols," in *Proceedings of the 2019 ACM Conference on Economics and Computation*, 2019, pp. 459–473.

[453] S. Lee and S. Kim, "Proof-of-stake at stake: predatory, destructive attack on pos cryptocurrencies," in *Proceedings of the 3rd Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, 2020, pp. 7–11.

[454] Y. Luo, Y. Chen, Q. Chen, and Q. Liang, "A new election algorithm for dpos consensus mechanism in blockchain," in *2018 7th International Conference on Digital Home (ICDH)*. IEEE, 2018, pp. 116–120.

[455] K. Wagner, T. Keller, and R. Seiler, "A comparative analysis of cryptocurrency consensus algorithms," in *Proceedings of the 16th International Conference on Applied Computing 2019*, 2019.

[456] L. Brünjes, A. Kiayias, E. Koutsoupias, and A.-P. Stouka, "Reward sharing schemes for stake pools," in *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2020, pp. 256–275.

[457] Leijurv, "leijurv/java-projects," Mar 2015. [Online]. Available: https://github.com/leijurv/Java-Projects/tree/master/RingSignatures

[458] A. Kak, "Lecture 8: Aes: The advanced encryption standard," *Lecture Notes on'Computer and Network Security', Purdue University, URL: https://engineering. purdue. edu/kak/-compsec/NewLectures/Lecture8. pdf*, 2016.

[459] S. Kallam, "Diffie-hellman: Key exchange and public key cryptosystems," *Master degree of Science, Math and Computer Science, Department of India State University, USA*, pp. 5–6, 2015.

[460] SpinResearch, "Spinresearch/rustysecrets," Aug 2018. [Online]. Available: https://github.com/SpinResearch/RustySecrets

[461] A. Ruiz-Alvarez and M. Humphrey, "A model and decision procedure for data storage in cloud computing," in *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)*. IEEE Computer Society, 2012, pp. 572–579.

[462] ——, "Toward optimal resource provisioning for cloud mapreduce and hybrid cloud applications," in *Proceedings of the 2014 IEEE/ACM International Symposium on Big Data Computing*. IEEE Computer Society, 2014, pp. 74–82.

[463] M. S. Yoon and A. E. Kamal, "Optimal dataset allocation in distributed heterogeneous clouds," in *2014 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2014, pp. 75–80.

[464] P. Plastiras and D. O'Sullivan, "Exchanging personal health data with electronic health records: A standardized information model for patient generated health data and observations of daily living," *International journal of medical informatics*, vol. 120, pp. 116–125, 2018.

[465] A. Cortez, P. Hsii, E. Mitchell, V. Riehl, and P. Smith, "Conceptualizing a data infrastructure for the capture, use, and sharing of patient-generated health data in care delivery and research through 2024 (white paper)," 2018.

[466] C.-F. Chung, K. Dew, A. Cole, J. Zia, J. Fogarty, J. A. Kientz, and S. A. Munson, "Boundary negotiating artifacts in personal informatics: Patient-provider collaboration with patient-generated data," in *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, 2016, pp. 770–786.

[467] R. J. Lordon, S. P. Mikles, L. Kneale, H. L. Evans, S. A. Munson, U. Backonja, and W. B. Lober, "How patient-generated health data and patient-reported outcomes affect patient–clinician relationships: A systematic review," *Health Informatics Journal*, p. 1460458220928184, 2020.

[468] G. Demiris, S. J. Iribarren, K. Sward, S. Lee, and R. Yang, "Patient generated health data use in clinical practice: a systematic review," *Nursing outlook*, vol. 67, no. 4, pp. 311–330, 2019.

[469] J. Feigenbaum, "Health insurance portability and accountability act," *Medicine, Retrieved from Semantic Scholar*, 2007.

[470] B. Bennett, T. Carney, M. Chiarella, M. Walton, P. Kelly, C. Satchell, and F. Beaupert, "Australia's national registration and accreditation scheme for health practitioners: A national approach to polycentric regulation," *Sydney L. Rev.*, vol. 40, p. 159, 2018.

[471] T. Mulder and M. Tudorica, "Privacy policies, cross-border health data and the gdpr," *Information & Communications Technology Law*, vol. 28, no. 3, pp. 261–274, 2019.

[472] K. Rantos, G. Drosatos, K. Demertzis, C. Ilioudis, and A. Papanikolaou, "Blockchain-based consents management for personal data processing in the iot ecosystem." in *ICETE (2)*, 2018, pp. 738–743.

[473] E. Harison, "Who owns enterprise information? data ownership rights in europe and the us," *Information & management*, vol. 47, no. 2, pp. 102–108, 2010.

[474] A. Albahri, A. Zaidan, O. Albahri, B. Zaidan, and M. Alsalem, "Real-time fault-tolerant mhealth system: Comprehensive review of healthcare services, opens issues, challenges and methodological aspects," *Journal of medical systems*, vol. 42, no. 8, p. 137, 2018.

[475] D. Isern and A. Moreno, "A systematic literature review of agents applied in healthcare," *Journal of medical systems*, vol. 40, no. 2, p. 43, 2016.

[476] V. Vaidehi, M. Vardhini, H. Yogeshwaran, G. Inbasagar, R. Bhargavi, and C. S. Hemalatha, "Agent based health monitoring of elderly people in indoor environments using wireless sensor networks," *Procedia Computer Science*, vol. 19, pp. 64–71, 2013.

[477] M. H. Record, "My health record," *Retrieved June 5, 2019 from https://www.myhealthrecord.gov.au/*, 2019.

[478] "Microsoft healthvault." [Online]. Available: https://www.healthvault.com/en-us/

[479] J. C. Mandel, D. A. Kreda, K. D. Mandl, I. S. Kohane, and R. B. Ramoni, "Smart on fhir: a standards-based, interoperable apps platform for electronic health records," *Journal of the American Medical Informatics Association*, vol. 23, no. 5, pp. 899–908, 2016.

[480] iOS Health Apple (Australia), "ios - health apple (australia)," *Retrieved June 5, 2019 from https://www.apple.com/au/ios/health/*, 2019.

[481] C. McFarlane, M. Beer, J. Brown, and N. Prendergast, "Patientory: A healthcare peer-to-peer emr storage network v1." *Entrust Inc.: Addison, TX, USA*, 2017.

[482] G. J. Katuwal, S. Pandey, M. Hennessey, and B. Lamichhane, "Applications of blockchain in healthcare: Current landscape & challenges," *arXiv preprint arXiv:1812.02776*, 2018.

[483] A. Hasselgren, K. Kralevska, D. Gligoroski, S. A. Pedersen, and A. Faxvaag, "Blockchain in healthcare and health sciences–a scoping review," *International Journal of Medical Informatics*, p. 104040, 2019.

[484] T. K. Mackey, T.-T. Kuo, B. Gummadi, K. A. Clauson, G. Church, D. Grishin, K. Obbad, R. Barkovich, and M. Palombini, "'fit-for-purpose?'–challenges and opportunities for applications of blockchain technology in the future of healthcare," *BMC medicine*, vol. 17, no. 1, p. 68, 2019.

[485] A. H. Mayer, C. A. da Costa, and R. d. R. Righi, "Electronic health records in a blockchain: A systematic review," *Health Informatics Journal*, p. 1460458219866350, 2019.

[486] M. Carter, "Introducing health information privacy in victoria," *Privacy Law and Policy Reporter*, vol. 7, no. 7, pp. 130–131, 2000.

[487] J. W. Coebergh, C. Van Den Hurk, S. Rosso, H. Comber, H. Storm, R. Zanetti, L. Sacchetto, M. Janssen-Heijnen, M. Thong, S. Siesling *et al.*, "Eurocourse lessons learned from and for population-based cancer registries in europe and their programme owners: improving performance by research programming for public health and clinical evaluation," *European journal of cancer*, vol. 51, no. 9, pp. 997–1017, 2015.

[488] "Australian cancer database (acd), retrieved from https://www.aihw.gov.au/about-our-data/our-data-collections/australian-cancer-database."

[489] C. Ringland, H.-T. Arkenau, D. O'Connell, and R. Ward, "Second primary colorectal cancers (spcrcs): experiences from a large australian cancer registry," *Annals of oncology*, vol. 21, no. 1, pp. 92–97, 2010.

[490] S. Y. Ko, K. Jeon, and R. Morales, "The hybrex model for confidentiality and privacy in cloud computing." *HotCloud*, vol. 11, pp. 8–8, 2011.

[491] H. Zhang, L. Ye, X. Du, and M. Guizani, "Protecting private cloud located within public cloud," in *Global Communications Conference (GLOBECOM), 2013 IEEE*. IEEE, 2013, pp. 677–681.

[492] D. Stantic and J. Jo, "Detecting abnormal ecg signals utilising wavelet transform and standard deviation," in *Proceedings of World Academy of Science, Engineering and Technology*, no. 71. World Academy of Science, Engineering and Technology (WASET), 2012, p. 208.

[493] A. Al Ghamdi and T. Thomson, "The future of data storage: A case study with the saudi company," *Journal of Electrical and Electronic Engineering*, vol. 6, no. 1, p. 1, 2018.

[494] Y. Yang and T. Chen, "Analysis and visualization implementation of medical big data resource sharing mechanism based on deep learning," *IEEE Access*, vol. 7, pp. 156 077–156 088, 2019.

[495] Y.-Y. L. Andy, C.-P. Shen, Y.-S. Lin, H.-J. Chen, A.-C. Chen, L.-C. Cheng, T.-F. Tsai, C.-T. Huang, L.-M. Chuang, and F. Lai, "Continuous, personalized healthcare integrated platform," in *TENCON 2012 IEEE Region 10 Conference*. IEEE, 2012, pp. 1–6.

[496] M. Peleg, Y. Shahar, S. Quaglini, A. Fux, G. García-Sáez, A. Goldstein, M. E. Hernando, D. Klimov, I. Martínez-Sarriegui, C. Napolitano *et al.*, "Mobiguide: a personalized and patient-centric decision-support system and its evaluation in the atrial fibrillation and gestational diabetes domains," *User Modeling and User-Adapted Interaction*, vol. 27, no. 2, pp. 159–213, 2017.

[497] V. I. Martinez, J. L. Marquard, B. Saver, L. Garber, and P. Preusse, "Consumer health informatics interventions must support user workflows, be easy-to-use, and improve cognition: applying the seips 2.0 model to evaluate patients' and clinicians' experiences with the conduit-hid intervention," *International Journal of Human–Computer Interaction*, vol. 33, no. 4, pp. 333–343, 2017.

[498] L. S. Liu, P. C. Shih, and G. R. Hayes, "Barriers to the adoption and use of personal health record systems," in *Proceedings of the 2011 iConference*, 2011, pp. 363–370.

[499] R. Hohemberger, C. E. da Roza, F. R. Pfeifer, R. M. da Rosa, P. S. S. de Souza, A. F. Lorenzon, M. C. Luizelli, and F. D. Rossi, "An approach to mitigate challenges to the electronic health records storage," *Measurement*, p. 107424, 2020.

[500] N. A. Busis, "How can i choose the best electronic health record system for my practice?" *Neurology*, vol. 75, no. 18 Supplement 1, pp. S60–S64, 2010.

[501] A. L. Weathers and G. J. Esper, "How to select and implement an electronic health record in a neurology practice," *Neurology: Clinical Practice*, vol. 3, no. 2, pp. 141–148, 2013.

[502] E. M. Hart, P. Barmby, D. LeBauer, F. Michonneau, S. Mount, P. Mulrooney, T. Poisot, K. H. Woo, N. B. Zimmerman, and J. W. Hollister, "Ten simple rules for digital data storage," *PLoS computational biology*, vol. 12, no. 10, 2016.

[503] G. Wilson, J. Bryan, K. Cranston, J. Kitzes, L. Nederbragt, and T. K. Teal, "Good enough practices in scientific computing," *PLoS computational biology*, vol. 13, no. 6, 2017.

[504] A. Boonstra and M. Broekhuis, "Barriers to the acceptance of electronic medical records by physicians from systematic review to taxonomy and interventions," *BMC health services research*, vol. 10, no. 1, p. 231, 2010.

[505] J. Ross, F. Stevenson, R. Lau, and E. Murray, "Factors that influence the implementation of e-health: a systematic review of systematic reviews (an update)," *Implementation science*, vol. 11, no. 1, p. 146, 2016.

[506] O. Ben-Assuli, "Electronic health records, adoption, quality of care, legal and privacy issues and their implementation in emergency departments," *Health Policy*, vol. 119, no. 3, pp. 287–297, 2015.

[507] S. I. Khan and A. S. M. L. Hoque, "Towards development of health data warehouse: Bangladesh perspective," in *2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT)*. IEEE, 2015, pp. 1–6.

[508] G. Kenny and R. Connolly, "Drivers of health information privacy concern: A comparison study," 2016.

[509] A. N. Joinson, U.-D. Reips, T. Buchanan, and C. B. P. Schofield, "Privacy, trust, and self-disclosure online," *Human–Computer Interaction*, vol. 25, no. 1, pp. 1–24, 2010.

[510] F. A. Rahim, Z. Ismail, and G. N. Samy, "A conceptual model for privacy preferences in healthcare environment," in *The 8th International Conference on Knowledge Management in Organizations*. Springer, 2014, pp. 221–228.

[511] C.-W. Chang, P. Liu, and J.-J. Wu, "Probability-based cloud storage providers selection algorithms with maximum availability," in *2012 41st International Conference on Parallel Processing*. IEEE, 2012, pp. 199–208.

[512] Z. ur Rehman, O. K. Hussain, S. Parvin, and F. K. Hussain, "A framework for user feedback based cloud service monitoring," in *2012 Sixth International Conference on Complex, Intelligent, and Software Intensive Systems*. IEEE, 2012, pp. 257–262.

[513] L. Qu, Y. Wang, and M. A. Orgun, "Cloud service selection based on the aggregation of user feedback and quantitative performance assessment," in *2013 IEEE International Conference on Services Computing*. IEEE, 2013, pp. 152–159.

[514] T. Halabi and M. Bellaiche, "Evaluation and selection of cloud security services based on multi-criteria analysis mca," in *2017 International Conference on Computing, Networking and Communications (ICNC)*. IEEE, 2017, pp. 706–710.

[515] ——, "A broker-based framework for standardization and management of cloud security-slas," *Computers & Security*, vol. 75, pp. 59–71, 2018.

[516] T. Halabi, M. Bellaiche, and A. Abusitta, "Online allocation of cloud resources based on security satisfaction," 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)*. IEEE, 2018, pp. 379–384.

[517] D. deBronkart and G. Eysenbach, "Gimme my damn data (and let patients help!): The# gimmemydamndata manifesto," *Journal of medical Internet research*, vol. 21, no. 11, p. e17045, 2019.

[518] T. McGhin, K.-K. R. Choo, C. Z. Liu, and D. He, "Blockchain in healthcare applications: Research challenges and opportunities," *Journal of Network and Computer Applications*, 2019.

[519] B. T. Rao *et al.*, "A study on data storage security issues in cloud computing," *Procedia Computer Science*, vol. 92, pp. 128–135, 2016.

[520] J. Sen, "Security and privacy issues in cloud computing," in *Architectures and protocols for secure information technology infrastructures*. IGI Global, 2014, pp. 1–45.

[521] N. A. Azeez and C. Van der Vyver, "Security and privacy issues in e-health cloud-based system: A comprehensive content analysis," *Egyptian Informatics Journal*, 2018.

[522] T. Trojer, B. Katt, T. Schabetsberger, R. Mair, and R. Breu, "The process of policy authoring of patient-controlled privacy preferences," in *International Conference on Electronic Healthcare*. Springer, 2011, pp. 97–104.

[523] I. H. Witten, E. Frank, L. E. Trigg, M. A. Hall, G. Holmes, and S. J. Cunningham, "Weka: Practical machine learning tools and techniques with java implementations," 1999.

[524] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, "Hybrid intrusion detection system based on the stacking ensemble of c5 decision tree classifier and one class support vector machine," *Electronics*, vol. 9, no. 1, p. 173, 2020.

[525] H. Yan, Z. J. Shi, and J.-H. Cui, "Dbr: depth-based routing for underwater sensor networks," in *International conference on research in networking*. Springer, 2008, pp. 72–86.

[526] A. Faz-Hernández, J. López, E. Ochoa-Jiménez, and F. Rodríguez-Henríquez, "A faster software implementation of the supersingular isogeny diffie-hellman key exchange protocol," *IEEE Transactions on Computers*, vol. 67, no. 11, pp. 1622–1636, 2017.

[527] M. A. Uddin, A. Stranieri, I. Gondal, and V. Balasubramanian, "An efficient selective miner consensus protocol in blockchain oriented iot smart monitoring," *The 20th IEEE International Conference on Industrial Technology, Melbourne, Australia*, 2019.

[528] O. Ali, A. Shrestha, J. Soar, and S. F. Wamba, "Cloud computing-enabled healthcare opportunities, issues, and applications: A systematic review," *International Journal of Information Management*, vol. 43, pp. 146–158, 2018.

[529] M. R. Mesbahi, A. M. Rahmani, and M. Hosseinzadeh, "Reliability and high availability in cloud computing environments: a reference roadmap," *Human-centric Computing and Information Sciences*, vol. 8, no. 1, p. 20, 2018.

[530] H. D. Zubaydi, Y.-W. Chong, K. Ko, S. M. Hanshi, and S. Karuppayah, "A review on the role of blockchain technology in the healthcare domain," *Electronics*, vol. 8, no. 6, p. 679, 2019.

[531] G. Zyskind, O. Nathan *et al.*, "Decentralizing privacy: Using blockchain to protect personal data," in *Security and Privacy Workshops (SPW), 2015 IEEE*. IEEE, 2015, pp. 180–184.

[532] F. Tian, "An agri-food supply chain traceability system for china based on rfid & blockchain technology," in *Service Systems and Service Management (ICSSSM), 2016 13th International Conference on*. IEEE, 2016, pp. 1–6.

[533] M. Samaniego and R. Deters, "Using blockchain to push software-defined iot components onto edge hosts," in *Proceedings of the International Conference on Big Data and Advanced Wireless Technologies*. ACM, 2016, p. 58.

[534] M. Aazam, P. P. Hung, and E.-N. Huh, "Smart gateway based communication for cloud of things," in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2014 IEEE Ninth International Conference on*. IEEE, 2014, pp. 1–6.

[535] P. K. Sharma, M.-Y. Chen, and J. H. Park, "A software defined fog node based distributed blockchain cloud architecture for iot," *IEEE Access*, vol. 6, pp. 115–124, 2018.

[536] Y. Tian and R. Hou, "An improved aomdv routing protocol for internet of things," in *Computational Intelligence and Software Engineering (CiSE), 2010 International Conference on*. IEEE, 2010, pp. 1–4.

[537] P. L. R. Chze and K. S. Leong, "A secure multi-hop routing for iot communication," in *Internet of Things (WF-IoT), 2014 IEEE World Forum on*. IEEE, 2014, pp. 428–432.

[538] O. Iova, F. Theoleyre, and T. Noel, "Using multiparent routing in rpl to increase the stability and the lifetime of the network," *Ad Hoc Networks*, vol. 29, pp. 45–62, 2015.

[539] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on wireless communications*, vol. 1, no. 4, pp. 660–670, 2002.

[540] O. Younis and S. Fahmy, "Heed: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Transactions on mobile computing*, vol. 3, no. 4, pp. 366–379, 2004.

[541] B. Diao, Y. Xu, Z. An, F. Wang, and C. Li, "Improving both energy and time efficiency of depth-based routing for underwater sensor networks," *International Journal of Distributed Sensor Networks*, vol. 11, no. 10, p. 781932, 2015.

[542] K. Latif, N. Javaid, A. Ahmad, Z. A. Khan, N. Alrajeh, and M. I. Khan, "On energy hole and coverage hole avoidance in underwater wireless sensor networks," *IEEE Sensors Journal*, vol. 16, no. 11, pp. 4431–4442, 2016.

[543] P. P. Jayaraman, X. Yang, A. Yavari, D. Georgakopoulos, and X. Yi, "Privacy preserving internet of things: From privacy techniques to a blueprint architecture and efficient implementation," *Future Generation Computer Systems*, 2017.

[544] B. Lee and J.-H. Lee, "Blockchain-based secure firmware update for embedded devices in an internet of things environment," *The Journal of Supercomputing*, vol. 73, no. 3, pp. 1152–1167, 2017.

[545] smhasher, *https://github.com/aappleby/smhasher, [Accessed Date:22 July, 2018]*.

[546] cityhash, *https://github.com/google/cityhash, [Accessed Date:22 July, 2018]*.

[547] B. Farahani, F. Firouzi, V. Chang, M. Badaroglu, N. Constant, and K. Mankodiya, "Towards fog-driven iot ehealth: Promises and challenges of iot in medicine and healthcare," *Future Generation Computer Systems*, vol. 78, pp. 659–676, 2018.

[548] D. Tosh, S. Shetty, P. Foytik, C. Kamhoua, and L. Njilla, "Cloudpos: A proof-of-stake consensus design for blockchain integrated cloud," in *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*. IEEE, 2018, pp. 302–309.

[549] M. U. Hassan, M. H. Rehmani, and J. Chen, "Privacy preservation in blockchain based iot systems: Integration issues, prospects, challenges, and future research directions," *Future Generation Computer Systems*, 2019.

[550] S. Moin, A. Karim, Z. Safdar, K. Safdar, E. Ahmed, and M. Imran, "Securing iots in distributed blockchain: Analysis, requirements and open issues," *Future Generation Computer Systems*, 2019.

[551] M. S. Ali, M. Vecchio, M. Pincheira, K. Dolui, F. Antonelli, and M. H. Rehmani, "Applications of blockchains in the internet of things: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, 2018.

[552] V. Balioti, C. Tzimopoulos, and C. Evangelides, "Multi-criteria decision making using topsis method under fuzzy environment. application in spillway selection," in *Multidisciplinary Digital Publishing Institute Proceedings*, vol. 2, no. 11, 2018, p. 637.

[553] Kass, "Programming blockchain," *.[Online]. Available: https://medium.com/programmers-blockchain [Accessed Date:19-April-2018].*

[554] R. Quinlan, "Data mining tools see5 and c5. 0," 2004.

[555] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: an update," *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[556] P. Gaži, A. Kiayias, and A. Russell, "Stake-bleeding attacks on proof-of-stake blockchains," in *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*. IEEE, 2018, pp. 85–92.

[557] P. Wei, Q. Yuan, and Y. Zheng, "Security of the blockchain against long delay attack," in *International Conference on the Theory and Application of Cryptology and Information Security*. Springer, 2018, pp. 250–275.

[558] A. Zhang, L. Wang, X. Ye, and X. Lin, "Light-weight and robust security-aware d2d-assist data transmission protocol for mobile-health systems," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 3, pp. 662–675, 2017.

[559] K. Biswas and V. Muthukkumarasamy, "Securing smart cities using blockchain technology," in *High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), 2016 IEEE 18th International Conference on*. IEEE, 2016, pp. 1392–1393.

[560] E. Mengelkamp, B. Notheisen, C. Beer, D. Dauer, and C. Weinhardt, "A blockchain-based smart grid: towards sustainable local energy markets," *Computer Science-Research and Development*, vol. 33, no. 1-2, pp. 207–214, 2018.

[561] J. Sun, J. Yan, and K. Z. Zhang, "Blockchain-based sharing services: What blockchain technology can contribute to smart cities," *Financial Innovation*, vol. 2, no. 1, p. 26, 2016.

[562] A. Stanciu, "Blockchain based distributed control system for edge computing," in *Control Systems and Computer Science (CSCS), 2017 21st International Conference on*. IEEE, 2017, pp. 667–671.

[563] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain technology: Beyond bitcoin," *Applied Innovation*, vol. 2, pp. 6–10, 2016.

[564] R. Neisse, G. Steri, and I. Nai-Fovino, "A blockchain-based approach for data accountability and provenance tracking," *arXiv preprint arXiv:1706.04507*, 2017.

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

[566] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *arXiv preprint arXiv:1702.05309*, 2017.

[567] S. Midya, A. Roy, K. Majumder, and S. Phadikar, "Multi-objective optimization technique for resource allocation and task scheduling in vehicular cloud architecture: A hybrid adaptive nature inspired approach," *Journal of Network and Computer Applications*, vol. 103, pp. 58–84, 2018.

[568] R. Shrestha and S. Kim, "Integration of iot with blockchain and homomorphic encryption: Challenging issues and opportunities," in *Advances in Computers*. Elsevier, 2019, vol. 115, pp. 293–331.

[569] J. Konečnỳ, H. B. McMahan, D. Ramage, and P. Richtárik, "Federated optimization: Distributed machine learning for on-device intelligence," *arXiv preprint arXiv:1610.02527*, 2016.

[570] H. Dang, T. T. A. Dinh, D. Loghin, E.-C. Chang, Q. Lin, and B. C. Ooi, "Towards scaling blockchain systems via sharding," in *Proceedings of the 2019 international conference on management of data*, 2019, pp. 123–140.