A Blockchain-based Hybrid Model for IoMT-enabled Intelligent Healthcare System

Ateeq Ur Rehman+, *Member IEEE,* Nargis Tariq+, *Student Member IEEE,* Mian Ahmad Jan, *Senior Member IEEE,* Fazlullah Khan*, *Senior Member IEEE,* Houbing Song, *Fellow IEEE,* and Muhammad Ibrahim, *Member IEEE,*

***Abstract*—In recent years, the healthcare industry has undergone a digital transformation, making patient data publicly available and accessible. Healthcare units make a portion of the data public while keeping the rest private, necessitating various mechanisms for security and privacy. Blockchain technology has been widely adopted in the healthcare sector to secure data transactions. However, public blockchains face challenges in scalability and privacy, whereas private blockchains struggle with centralization, interoperability, and complexity. To address these challenges, we propose an Internet of Medical Things (IoMT)-based hybrid blockchain architecture. The proposed architecture combines the decentralized Ethereum and the centralized Hyperledger Fabric blockchain (Eth-Fab) using SQLite to leverage Ethereum smart contracts with the Hyperledger permission model. Moreover, we introduce access control strategies to enhance patient data authentication and authorization. We have employed machine learning algorithms to assist healthcare practitioners in accurately detecting diseases and making time-efficient decisions. Additionally, we modeled the proposed architecture using the M/M/1 queuing model and derived closed-form expressions for latency, throughput, and server utilization. The validity of these expressions was verified through Monte Carlo simulations. The results demonstrate that higher service times (block generation) yield better outcomes in terms of latency, throughput, and utilization, regardless of the arrival time, i.e., transactions in the mining pool.**

***Index Terms*—Internet of Medical Things, Hyperledger Fabric, Ethereum, Security, privacy, Machine Learning, M/M/1 Queuing Model, Monte Carlo.**

I. Introduction

INTERNET of Medical Things (IoMT) and machine learning (ML) have become integral in the healthcare industry, providing a sophisticated platform for the interaction of various entities. IoMT facilitates global access to hospitals for online healthcare services and data processing. The industry is anticipated to reach a market value of $390.7 billion USD by 2024 [1]. IoMT generates an immense volume of medical data, which can be collected and analyzed swiftly and cost-effectively using ML algorithms. Additionally, real-time data processing in IoMT is enhanced by the availability of smart wearables in the market [2]. However, concerns have been raised regarding the security of these wearables, as they may not offer sufficient protection for patients' health records [3].

Recent statistics from the United States have reported about 20,030 health-related data breaches, impacting approximately 199.3 million records [4]. These statistics highlight the significant risk to healthcare data, underscoring the necessity to investigate vulnerabilities and develop effective solutions.

In conventional IoMT-enabled healthcare systems, centralized server-based solutions are predominantly employed for storing and sharing health information, where security measures rely on the assumption that the IoMT server is trustworthy [5]. This approach, however, introduces a single point of failure; if compromised, it can jeopardize the integrity of the entire system. Blockchain technology has emerged as a prominent solution for securing and preserving healthcare data, with both public and private (permission-based) blockchains being widely adopted. Public blockchains, such as those based on Ethereum, offer transparency but face challenges in scalability and privacy due to their open network structure. Conversely, permission-based blockchains like Hyperledger offer a more controlled environment but struggle with issues of centralization and complexity.

Notable efforts in the literature include the work by the authors in [6], who proposed a method for secure access, storage, and sharing of patient health information using Ethereum and the Interplanetary File System (IPFS). Additionally, [7] utilized Ethereum, Django, and IPFS for managing medical records, while [8] combined real-time data and predictive analytics with Hyperledger Fabric to facilitate data processing, analysis, and storage for veterinary clinic data, ensuring access is limited to authorized participants. Despite these advancements, most existing solutions focus either on public or private blockchains, each with distinct limitations. For instance, private blockchains restrict data access to specific nodes, limiting participant numbers and potentially allowing for higher throughput and faster consensus. Public blockchains, however, suffer from slower transaction processing times, which may deter organizations from making their data publicly available [9]. Decentralized cryptographic primitives have been suggested to increase transaction speed while reducing costs, making them suitable for IoMT-enabled healthcare systems [10]. Merging public and permissioned blockchains could sig-

Ateeq Ur Rehman is with the Department of Computing, Staffordshire University, United Kingdom. E-mail:{ ateequr.rehman@staffs.ac.uk

N. Tariq and M. Ibrahim are with the Department of Information Technology, University of Haripur, Pakistan. E-mail:{nargis.tariq, m.ibrahim}@uoh.edu.pk

M. A. Jan is with the Department of Computer Science, College of Computing and Informatics, University of Sharjah, Sharjah, 27272, UAE Email: mjan@sharjah.ac.ae

F. Khan is with the School of Computer Science, University of Nottingham Ninbo China, Ningbo 315104, Zhejiang, China. E-mail: fazl.ullahg@nottingham.edu.cn

Houbing Song is with the College of Engineering and Information Technology, University of Maryland, United States. E-mail: songh@umbc.edu

+ shows first-equal authors and * shows corresponding author.

nificantly enhance data integrity within the healthcare system.

In this paper, we propose an IoMT-based hybrid blockchain architecture, integrating the decentralized Ethereum with the centralized Hyperledger Fabric blockchain (Eth-Fab) using SQLite. This Eth-Fab model fetches the patient's iris image using IoMT sensors, stores it on IPFS, and secures its data hash using Ethereum. We employ IPFS, Ethereum, Hyperledger Fabric with SQLite, and hybrid access control policies to ensure patients can access their disease records using smart devices without needing to visit the hospital. Additionally, machine learning algorithms such as multi-layer perceptron (MLP), support vector machine (SVM), extra trees classifier (ETC), and stochastic gradient descent (SGD) are utilized for disease prediction, with the results securely stored on the blockchain, accessible to patients and partially to healthcare practitioners. Our proposed model not only secures health data but is also analytically modeled using the M/M/1 queuing model, deriving closed-form expressions for latency, transaction validation, throughput, and system utilization. This model achieves high transaction speed and low cost, validated through Monte Carlo simulation, showing decreased latency with increased service time and improved throughput with increased arrival and service times. Utilization decreases with the rise in arrival and service times, demonstrating the model's efficiency.

The primary contributions of our research are summarized as follows:

1) We introduce a novel hybrid model for securing the Internet of Medical Things (IoMT) data that combines Ethereum and Hyperledger Fabric alongside SQLite. This model ensures patient privacy through a permissioned blockchain framework, enhanced by newly designed access control mechanisms. This approach enables secure patient registration via a modified smart contract.
2) Our architecture aims to safeguard the privacy and security of patient data by employing hybrid access control techniques. These techniques restrict data access to authorized personnel only, thereby mitigating the risk of data breaches and unauthorized access.
3) We leverage machine learning algorithms for disease prediction and then securely storing on the blockchain, ensuring data integrity and accessibility.
4) We adopt the M/M/1 queuing model to formulate our proposed architecture, from which we derive closed-form expressions for key performance metrics such as latency, throughput, and system utilization. The efficacy of these expressions is further validated through Monte Carlo simulation.

The rest of the paper is organized as follows. Section II describes the proposed system model followed by a proof of concept utilizing mathematical modeling in Section III. Section IV describes experimental assessment, and Section V brings the investigation to the conclusion.

II. System Model

The proposed model leverages various technologies to achieve its objectives. Ethereum is utilized for registration, enabling IoMT devices to optionally provide their public keys, passwords, document hashes, and EHRs Manager, among others. Hyperledger Fabric is employed for storing patients' data, while SQLite facilitates the connection between Hyperledger Fabric and Ethereum. In our hybrid blockchain, we employ Practical Byzantine Fault Tolerance (pBFT) consensus on the Hyperledger Fabric side to ensure trust and validation through known and authorized participants. Conversely, proof of work (PoW) is used on the Ethereum side as the underlying consensus mechanism, ensuring the network's decentralized and trustless nature. Through the integration of pBFT and PoW, private and public blockchains are merged, allowing miners to collect pending transactions and solve the complex PoW challenge. The first miner to solve the challenge broadcasts the solution and the new block of transactions to the network. Other network users verify the solution and transactions, embodying trust through competition. This integration of technologies ensures accurate disease prediction while maintaining safe and efficient data handling. Furthermore, machine learning techniques are applied to predict illnesses based on the available data. Figure 1 illustrates the system architecture, which is elaborated upon in the subsequent subsections.

A. Biometric Iris Scanners

In the proposed model, IoMT-based iris scanners enhance patient identification security and robustness through the following steps:

- Eye Image Capture: The iris scanner captures eye images for biometric identification via iris recognition or eye-tracking technologies.
- Iris Segmentation: The scanner segments the iris for characterization, detecting the pupil.
- Iris Normalization: Focuses on the pupillary side of the eye.
- Feature Extraction: Segmentation and normalization are employed to augment a normal iris image, achieving a well-distributed picture.
- Data Encoding: Converts data types.
- InterPlanetary File System (IPFS): Stores encrypted data to generate a hash.

B. Proposed Working Principle of Ethereum

Ethereum facilitates patient registration through public key document hashes, passwords, and additional information (e.g., the EHR manager's desk number and name). To preserve patients' and practitioners' information, the following steps are undertaken:

1) Patient Registration: Let P denote the set of patients, where each patient is assigned a public key, document hash H, password PW, manager desk number MDN, and name MN, represented as tuples (H, PW, MDN, MN). The registration process, modeled as a function R, maps patient information to Ethereum addresses:

$$R : (H, PW, MDN, MN) \rightarrow E \tag{1}$$

where E is the set of Ethereum addresses.

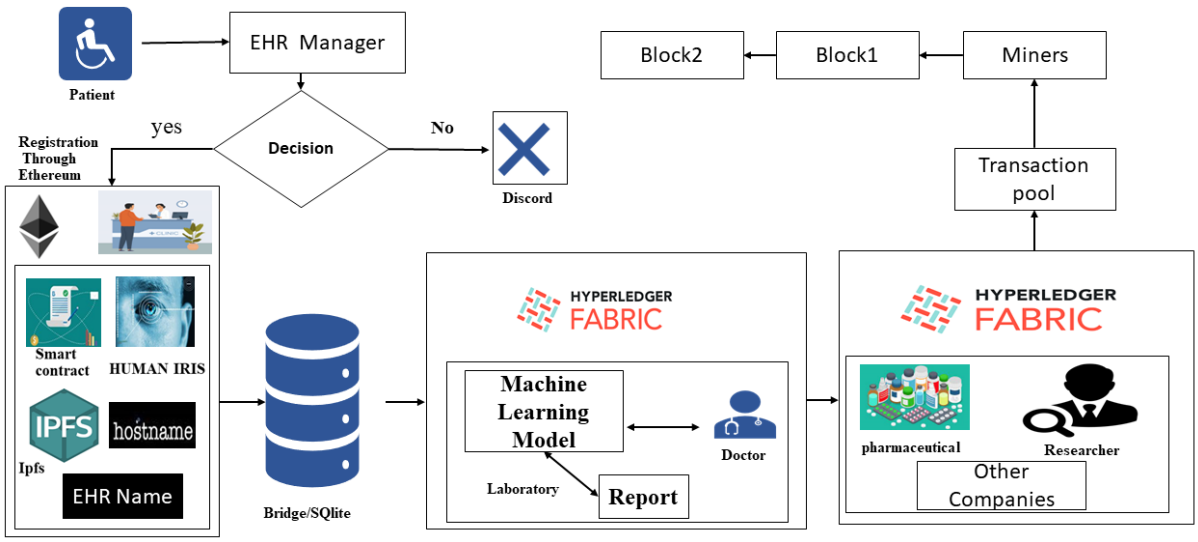

Fig. 1. System Architecture demonstrating the use of Ethereum and Hyperledger Fabric with SQLite to forecast disease data using Machine Learning

2) Smart Contract Automation: Let C represent healthcare processes automated via smart contracts. Each process in C accepts inputs I and produces outputs O. Ethereum smart contracts execute these functions automatically, enhancing healthcare system efficiency.

$$M_L = \left(\frac{i}{R}\right) \times \sum(|IO|) \tag{2}$$

Here, R extracts patient information from Eq. (1), with $i \geq 1$ denoting the number of patients. M_L calculates the mean length of patient data for estimating blockchain-stored patient data.

C. Proposed Working Principle of Hyperledger Fabric

This section focuses on using Hyperledger Fabric to store patients' disease data P_d, granting patients more control over their medical information and enabling healthcare providers to make informed treatment decisions T_d. The following actions safeguard user data and patient health information, formulated in Eq. (3):

1) HF Channels (H_c): Secure communication channels among patients, doctors, pharmaceuticals, and hospital managers are established.
2) Membership Service Provider (MSP) manages identities on the blockchain, authenticating new patients via the EHR manager.
3) Committing Nodes (CN) and Endorsing Nodes (EN) execute and verify transactions, respectively.
4) Ordering Nodes (ON) maintain transaction sequences.
5) Chaincode (CC) implements smart contracts for interfacing with the distributed ledger.
6) Certificate Authority (CA) issues certificates, adding patient information to the blockchain.

Treatment decisions T_d are derived as follows:

$$T_d = f(P_d, H_c, MSP, CN, EN, ON, CC, CA) \times C_l \tag{3}$$

The function f processes and secures P_d on Hyperledger Fabric, considering all input parameters. Commitment level C_l, ranging from 0 to 1, indicates data integrity and security levels.

1) Transaction Flow: Transactions involve two clients, the patient and the doctor, relying on each other for health assessments. An endorsement policy requires both parties to endorse any transaction, involving several stages for completion.

1) The doctor initiates a transaction.
2) Endorsing peers verify the signature and transaction execution.
3) Proposal responses are inspected.
4) Target peers assemble endorsements into a transaction.
5) Transactions are validated and committed.
6) Update the ledger.

D. Machine Learning-based Diseases Prediction

Machine learning has emerged as a significant tool in healthcare, with the potential to transform medical practice and enhance patient outcomes. Machine learning may be used in a variety of activities in healthcare, including illness detection, risk classification, therapy selection, and patient monitoring. One of the most significant advantages of machine learning in healthcare is its ability to manage large and complex datasets, which are commonly encountered in medical practice. Malaria, diabetes, impetigo, diabetes, AIDS, jaundice, chickenpox, and other health disorders have a significant influence on one's health and, if untreated, can lead to death [11]. In this situation, data mining approaches such as SVM, multi-layer perception, stochastic gradient descent (SGD), and additional tree classifier algorithms might be useful. We employed the aforementioned algorithms to detect the diseases of the patient.

E. Access Control Mechanism

Access control policies (ACP) are a collection of rules that determine who has access to resources, and what resources

should they have access to. These ACPs are unique to the organization and must be kept private. Role Base Access Control (RBAC) and Attribute-based Access Control (ABAC) ACPs are fully role-centric and subject-centric. They do not enable management to monitor the user's compliance with work requirements or to assess the organization's aims [12]. Healthcare systems manage sensitive and personal data, which necessitates tight access restrictions to preserve patient privacy and data security. We proposed ABAC and RBAC policies for a healthcare system that uses hybrid blockchain technology, machine learning, and secure data access in this study.

1) Attribute-based Access Control Model: The ABAC policy considers user, resource, and environmental factors, defining access control rules through logical formulas. This policy facilitates fine-grained access control, adaptable to the diverse roles and requirements within the healthcare sector. Attributes of resources include their sensitivity level, data type, and location, while environmental attributes cover the time of day and network address. Access control rules are established using logical formulas, enabling precise control over access rights.

The design of the ABAC policy emphasizes flexibility, allowing it to be customized for different roles and needs within the healthcare system. Consequently, system elements such as users (U), objects (O), sessions (S), and actions (OP), along with user characteristics (A_u), object attributes (A_o), session attributes (A_s) and environmental attributes (A_e) follow mappings specified by NIST for subject attributes, object attributes, and environmental characteristics, as detailed in the guide by [13]. Additionally, some definitions are inspired by [14], further enriching the framework's conceptual underpinnings.

$$A_{total} = A_u \cup A_o \cup A_s \cup A_e, \tag{4}$$

A_{total} is the total set of attributes considered in the ABAC policy. This equation encapsulates the essence of ABAC by highlighting the importance of a multifaceted attribute set in the decision-making process for access control, ensuring that access rights are precisely granted based on a wide range of factors.

Definition 1 (Range of Attributes)**.** *Given a property $a \in A_e$, the attribute set V_a is the set of all permissible values in the system.*

Definition 2 (Function of an Attribute)**.** *Given an entity $e \in A_e$, a characteristic function f_{ae} is a function that maps an entity to a specific attribute range value. $f_{ae}(e, a)$ specifically returns the value of attribute a for entity e.*

EXAMPLES

a) Example 1: Doctors (D_s) may access patient records if those patients have provided permission. Only users with the necessary professional qualifications are allowed to access sensitive patient data.

b) Example 2: Nurses can only access patient records for the ward to which they are assigned. This means they cannot view records for patients not in their ward, nor can they view records for patients in other wards.

c) Example 3: Patients can only access their own records, and these records can only be accessed from authorized locations.

Definition 3 (Attribute Expressions)**.** *Let R_t, R_o, and P_r represent the sets of Resource types, User roles, and Patient Records, respectively, for access control based on attribute expressions. If $U = D_s$ and $R_t = P_r$, then access is allowed.*

Definition 4 (Filter by Attribute)**.** *Let N_s, A_w, and W_0 represent the sets of Nurses, their assigned Wards, and Ward locations, respectively. If $U = N_s$ and $R_t = P_r$ and $A_w = W_0$, then access is allowed.*

Definition 5 (Satisfaction with Attribute Filters)**.** *Let P_a, R_o, and U_i represent the sets of Patients, Resource owners, and User IDs, respectively. If $U = P_a$ and $R_o = U_i$, then access is allowed.*

Definition 6 (Satisfaction with a Relationship Condition)**.** *An entity $e \in A_e$ fulfills an attribute filter F, indicated by the symbol $e = F_{ae}$, if and only if*

$$\forall (a_i, v_i) \in F, f_{ae}(e, a_i) = v_i, \tag{5}$$

$$\textit{and} \quad \forall (a_i, \neg v_i) \in F, f_{ae}(e, a_i) \neq v_i. \tag{6}$$

Eq. (5) *and (6) collectively define the conditions under which an entity satisfies an attribute filter within the ABAC framework*

Definition 7 (Access Request)**.** *An access request is represented by the tuple $q = (u, o, s, o_{pi})$ where user $u \in U$ requests that operation op be performed by the system $\in OP$ on object $o \in O$ in session $s \in S$.*

Definition 8 (Rule Satisfaction)**.** *An access request $q = (u, o, s_i, o_{pi})$ is said to satisfy a rule ρ, denoted as $q = \rho$, if and only if*

$$(u, o, s_i) = F \wedge (u, o, s_i) = R \wedge o_{pi} = op_{\rho}. \tag{7}$$

Definition 9 (ABAC Policy)**.** *An ABAC policy is a tuple $\pi = (E, OP, A, f_{ae}, P_i)$ where E, OP, A, and P represent the sets of entities, actions, attributes, and ABAC rules in the system, respectively. The attribute function is denoted by f_{ae}.*

Definition 10 (ABAC Policy Decision)**.** *The decision of an ABAC policy $d\pi(q)$ for an access request q is permitted if and only if*

$$\exists \rho \in \pi : d\pi(q) = \rho. \tag{8}$$

Otherwise, the decision will be to deny access. The system's decision is authorized if an access request satisfies a rule of the access control policy. If the access request violates any of the rules in the access control policy, the system will reject it.

2) Role-based Access Control Model (RBAC): The RBAC concept has three main components roles, permissions, and users. The EHR defines roles and permission for users to perform certain activities within the system. For example, a doctor may be assigned the role of "physician", which allows him to examine patient records. However, a nurse may be assigned the role of "nurse", which allows her to view patient

records who are under her care. RBAC has several advantages, including increased security, streamlined administration, and a lower chance of human mistakes.

Theorem 1. *Let U denote the collection of system users, O the set of objects (for example, patient records) that users may access, and R the set of roles to which users can be assigned. We define a permission function $P : U \times O \rightarrow \{0, 1\}$, which maps each user and object pair to a binary value indicating whether the user has access to the object.*

Proof. Each user is allocated to one or more roles under RBAC management, with each role associated with a specific set of permissions. The system administrator defines these permissions for each role based on the users' job duties and their need-to-know basis. Access to objects is granted to users based on their assigned roles, rather than their individual identities. Representing this system as a graph where each node symbolizes a user, object, or role, and each edge denotes a permission relationship between a user and an object, or a role and an object, we utilize graph theory to demonstrate that RBAC can effectively restrict access to patient data, thereby safeguarding privacy and security. □

Theorem 2. *Let $G = (V, E)$ represent the access control system graph, where V denotes nodes and E denotes edges. A cut (S, T) of the network is defined as a partition of the nodes into two disjoint sets S and T, such that S includes the source node (the user seeking access) and T contains the destination node (the object being accessed). A cut is considered (s, t)-valid if $s \in S$, $t \in T$, and there exists at least one path in the graph from s to t that respects the permission constraints defined by the edges in E.*

Proof. In the context of RBAC, permission relationships between users and objects are represented as edges in the graph G. To ascertain whether a user u is permitted access to an object o, it is necessary to determine if there exists an (u, o)-valid cut in the graph. The existence of such a cut implies a permissible path from u to o within G, thereby granting u access to o. Conversely, the absence of an (u, o)-valid cut indicates that u is not authorized to access o. This framework demonstrates the efficacy of RBAC in safeguarding patient data, ensuring privacy, and enhancing security. By structuring access permissions through roles and delineating access pathways within a graph model, healthcare systems can meticulously regulate access to sensitive information, effectively mitigating the risk of unauthorized disclosure. □

F. SQLite

SQLite is a self-contained library that implements a SQL database engine without requiring a server process, distinguishing it from most other SQL databases. It is renowned for its exceptional storage efficiency, rapid query execution, support for ACID (Atomicity, Consistency, Isolation, Durability) transactions, and minimal memory requirements. Furthermore, SQLite simplifies the deployment process by eliminating the need for setup or ongoing management [15]. In our project, SQLite serves as an intermediary facilitating the transfer of patients' public keys between Ethereum and Hyperledger Fabric. It is designed to temporarily store data, ensuring that any public key transmitted across these hybrid blockchains is automatically purged from SQLite upon power loss, thereby precluding the possibility of third-party intervention.

III. Proof of Concept using Mathematical Modelling

To assess the performance of blockchain with machine learning, we used the M/M/1 Queuing Model.

A. M/M/1 Queuing-based Mathematical Modelling

The M/M/1 Queuing Model is a prominent mathematical model utilized to assess the performance of queuing systems. It is denoted as M/M/1, signifying Markovian (memoryless) arrival processes, Markovian service processes, and a single server. This model applies to systems where both the arrival rate of customers (λ) and the service rate of the server (μ) are constant over time [16].

The M/M/1 model facilitates the computation of various performance metrics, such as the average number of customers in the system, the average time customers spend waiting in the queue, the probability that the system is empty, and the probability that it is full. Key performance measures derived from the model include: The average number of customers in the system (L):

$$L = \frac{\lambda}{\mu - \lambda} \tag{9}$$

The average time a customer spends in the queue (W):

$$W = \frac{1}{\mu - \lambda} \tag{10}$$

In these equations, λ represents the rate of customer arrivals, and μ denotes the service rate of the server (customers served per unit time).

B. M/M/1 Queuing-based Mathematical Modelling

The M/M/1 Queuing Model is a well-known mathematical model for evaluating the effectiveness of queuing systems. M/M/1, or Markovian arrival, Markovian service, and one server, refers to systems with a single server in which both the customer arrival rate and the server service rate stay constant throughout time. [16]. The M/M/1 model may be used to create a variety of performance metrics, such as the average number of customers using the system, the average amount of time consumers wait in line, the likelihood that the system is empty, and the likelihood that it is full. The following is the formula for various performance measures: The typical client count in the system (L):

$$L = \frac{\lambda}{(\mu - \lambda)} \tag{11}$$

The average length of time spent in line (W):

$$W = \frac{1}{(\mu - \lambda)} \tag{12}$$

In these Equations, λ and μ indicate the server's service rate and the client arrival rate (customer time per unit).

C. Enrollment Stage

The proposed research study aims to establish a secure and efficient platform for managing patient data by integrating blockchain technology, machine learning, and secure data access mechanisms. The initial phase involves patient registration on the Ethereum blockchain, where patients create a unique identity. This identity, which includes patient information, facilitates authentication. The registration process employs public keys to ensure that only authorized individuals can access the data.

1) Operating Characteristics: The dynamics of EHR systems can be analyzed using queuing theory, with the following probabilities defined: The probability that EHRs are busy, denoted by P_b:

$$P_b = \frac{\lambda}{\mu} \tag{13}$$

The probability that EHRs are idle, represented by P_0:

$$P_0 = 1 - \frac{\lambda}{\mu} \tag{14}$$

The probability of having n EHRs in use within the hospital, P_n:

$$P_n = \left(\frac{\lambda}{\mu}\right)^n \times P_0 \tag{15}$$

The probability that the number of patients exceeds a specified number K:

$$P(n > K) = \left(\frac{\lambda}{\mu}\right)^{K+1} \tag{16}$$

The probability that the queue is non-empty:

$$P(n > 1) = 1 - P_0 - P_1 = \left(\frac{\lambda}{\mu}\right)^2 \tag{17}$$

The expected number of patients in the system, L_s, is given by:

$$L_s = \frac{\lambda}{\mu - \lambda} \tag{18}$$

The average number of patients in the queue, L_q, should be:

$$L_q = L_s - \frac{\lambda}{\mu} = \frac{\lambda^2}{\mu(\mu - \lambda)} \tag{19}$$

The average waiting time for Ethereum registration to be completed, W_s, can be computed as:

$$W_s = \frac{L_s}{\lambda} = \frac{1}{\mu - \lambda} \tag{20}$$

D. Transmit Stage

The second stage involves utilizing Hyperledger Fabric to store patient disease data securely. This data is accessible only to authorized individuals, ensuring the privacy and security of patient information. Providers can connect to the hospital network and access essential data to deliver appropriate medication-related information. Hyperledger Fabric ensures that patient data is safe, immutable, and tamper-proof, while also facilitating easy data access and exchange among healthcare practitioners.

To calculate L_n, which represents the average time a doctor spends writing data on Hyperledger Fabric, we have:

$$L_n = \frac{\mu}{\mu - \lambda} \tag{21}$$

The average latency W using the M/M/1 queuing model is given by:

$$W = \frac{1}{\mu - \lambda} \tag{22}$$

The average throughput X can be computed as:

$$X = \lambda \times (1 - \frac{\lambda}{\mu}) \tag{23}$$

The system utilization ρ using the M/M/1 queuing model is:

$$\rho = \frac{\lambda}{\mu} \tag{24}$$

Theorem 3. *Let λ be the arrival rate of patient registration requests, and μ be the service rate of the system. The proposed research integrates blockchain technology, machine learning, and secure data access to develop a safe and efficient platform for handling patient data, represented by an M/M/1 queuing model. In this model:*

- *$\rho = \frac{\lambda}{\mu} < 1$ represents the system's utilization,*
- *if $\rho > 1$, the system is overloaded, leading to an infinitely growing queue,*
- *if $\rho < 1$, the system can efficiently manage the queue, ensuring it remains finite.*

Proof. Let L_q represent the average number of patients in the queue and L_{ss} the average number of patients serviced by the system. Let W_q be the average queue waiting time, and W_s the average service time. Let W represent the overall average time a patient spends in the system.

The arrival rate λ and the service rate μ are foundational to calculating the system's performance metrics using the M/M/1 queuing model. Little's Law, $L = \lambda W$, provides a relationship between the average number of items in the system (L), the arrival rate (λ), and the average time spent in the system (W). Utilizing this, along with the utilization factor ρ, allows for comprehensive modeling of the system's dynamics, demonstrating the effectiveness of the proposed platform for secure patient data management. □

Theorem 4. *Let λ be the arrival rate of patient registration requests, and μ be the service rate of the system. This research proposes integrating blockchain technology, machine learning, and secure data access to develop a safe and efficient platform for handling patient data, modeled by an M/M/1 queuing system. In this model:*

- *$\rho = \frac{\lambda}{\mu} < 1$ signifies the system's utilization is below its capacity, enabling efficient data processing.*
- *If $\rho > 1$, the system becomes overloaded, leading to an indefinitely growing queue.*
- *If $\rho < 1$, the system efficiently processes requests, maintaining a manageable queue length.*

Proof. Let L_q denote the average number of patients waiting in the queue and L_{ss} the average number of patients being serviced. Let W_q represent the average queue waiting time and

W_s the average service time. Let W signify the total average time a patient spends in the system.

Given:

$$\lambda = \frac{L_s + L_q}{W_s} \tag{25}$$

Where λ is derived from the M/M/1 model specification, indicating the arrival rate of patient registration requests.

The service rate μ is determined as:

$$\mu = \frac{1}{W_s} \tag{26}$$

Utilizing Little's Law, we find:

$$L = \lambda W \tag{27}$$

Where L is the average number of patients in the system.

The average number of serviced patients L_{ss} is influenced by the system's utilization ρ, thus:

$$L_{ss} = \rho L \tag{28}$$

The average queue length L_q is then recalculated to reflect the time patients spend waiting minus the time being serviced, as:

$$L_q = \lambda(W - W_s) \tag{29}$$

Simplifying, we get:

$$L_q = \lambda(1 - \rho)\frac{1}{\mu} \tag{30}$$

For W, the average time a patient spends in the system, applying Little's Law yields:

$$W = \frac{L_{ss} + L_q}{\lambda} \tag{31}$$

Simplifying further:

$$W = \frac{\rho L + \lambda(1 - \rho)\frac{1}{\mu}}{\lambda} \tag{32}$$

This simplifies to:

$$W = \frac{1}{\mu} \tag{33}$$

indicating that the average time in the system is inversely related to the service rate.

Lastly, the average waiting time in the queue W_q is:

$$W_q = \frac{L_q}{\lambda} \tag{34}$$

Leading to:

$$W_q = (1 - \rho)\frac{1}{\mu} \tag{35}$$

This demonstration of Little's Law for the M/M/1 queuing model underscores the relationship between system utilization, service rates, and waiting times, affirming the model's applicability to managing patient data in a blockchain-integrated healthcare system. □

IV. Experimental Evaluation

The analytical approach integrates collected performance metrics into the Hyperledger Fabric blockchain framework. This process involves utilizing the M/M/1 queuing model, parameterized from the experimental testbed, to structure our model. To substantiate the proposed model's validity, we juxtapose the analytical outcomes with results derived from Monte Carlo simulations, facilitating a comprehensive evaluation of the model's performance in simulating real-world operational scenarios.

A. Experimental Setup

We have successfully developed an Ethereum blockchain application using Solidity. For Hyperledger implementation, TypeScript, the Web3 API, and SQLite queries were employed, all integrated within a Node.js and JavaScript environment. Python was utilized for the system's machine learning components. The entire system operates efficiently on the Windows 11 operating system, supported by Docker Desktop and Visual Studio Code for development and testing.

B. Monte Carlo Simulation

Monte Carlo simulation is a computational technique used to model and simulate the behavior of complex systems or processes. It employs statistical methods, utilizing random sampling and probability distributions to estimate the potential outcomes of a system or process [17]. This approach generates a large number of random samples to represent various possible outcomes, which are then analyzed to compute the probability distribution of those outcomes. The results facilitate estimating expected values and assessing the risks associated with different scenarios [17].

a) Monte Carlo Formula for Latency: The formula for calculating average latency using Monte Carlo simulation is given by:

$$\text{Latency} = \frac{1}{n}\sum_{i=1}^{n}(D_i - A_i) \tag{36}$$

where n represents the number of jobs, A_i is the arrival time of the i-th job, D_i is the departure time of the i-th job, and the summation $\sum_{i=1}^{n}(D_i - A_i)$ calculates the total latency for all n jobs.

b) Monte Carlo Formula for Throughput: Throughput is calculated as:

$$\text{Throughput} = \frac{\text{num}_{\text{trials}}}{\text{enter}_{\text{time}} + \text{service}_{\text{time}}} \tag{37}$$

where $\text{num}_{\text{trials}}$ is the number of trials conducted during the simulation, $\text{enter}_{\text{time}}$ is the time at which the first job entered the system, and $\text{service}_{\text{time}}$ is the total time spent serving all jobs.

c) Monte Carlo Formula for Utilization: Utilization is determined by:

$$\text{Utilization} = \text{mean}(\mu) \times \lambda \tag{38}$$

where $\text{mean}(\mu)$ represents the average service rate, and λ is the arrival rate of jobs to the system.

C. Average Latency

This subsection elaborates on the average latency performance metric, analyzed through both analytical modeling using the M/M/1 model and simulation via the Monte Carlo method. Latency, or response time, includes the total time a customer spends in the system, encompassing both waiting in the queue and being serviced by the server. In our proposed model, latency specifically measures the duration required for a transaction to be processed and mined into a block. Fig.

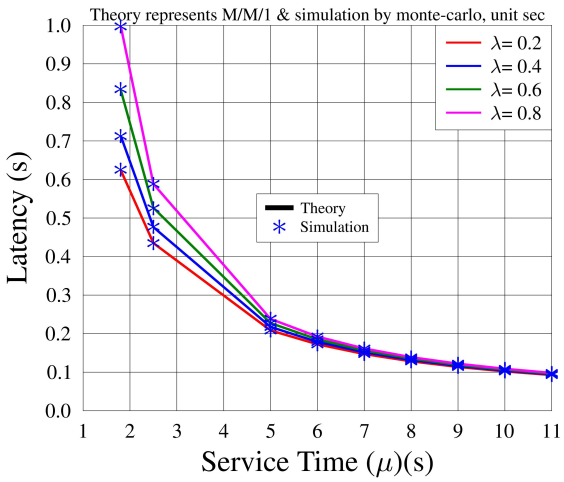

Fig. 2. Latency Performance in light of Service Time and Arrival Rate.

2 illustrates that at $\lambda = 0.2$ and $\mu = 1.8$, the latencies for Monte Carlo and M/M/1 are 0.62561 seconds and 0.625 seconds, respectively. Similarly, at $\lambda = 0.8$ and $\mu = 11$, the Monte Carlo result is 0.098075513, and the M/M/1 result is 0.098039216. This validates the accuracy of the equations derived through M/M/1 for latency. Furthermore, the figure shows that increasing the μ decreases the average latency as a higher μ enables the server to process customers faster, reducing the queue length. However, latency begins to rise again as μ significantly exceeds λ, indicating server underutilization.

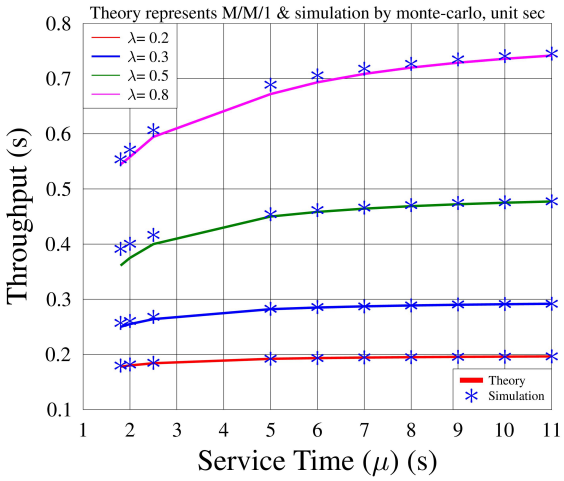

Fig. 3. Throughput Performance in Light of Service Time and Arrival Rate

D. Average Throughput

This subsection examines the throughput performance of our proposed model using both M/M/1 and Monte Carlo simulations. Throughput, the rate at which the system processes and completes customer requests, reflects the system's efficiency in handling transactions, equivalent to the customer arrival rate. Fig. 3 demonstrates that throughput increases with the λ and μ, showcasing the model's capability to process transactions faster as service time increases.

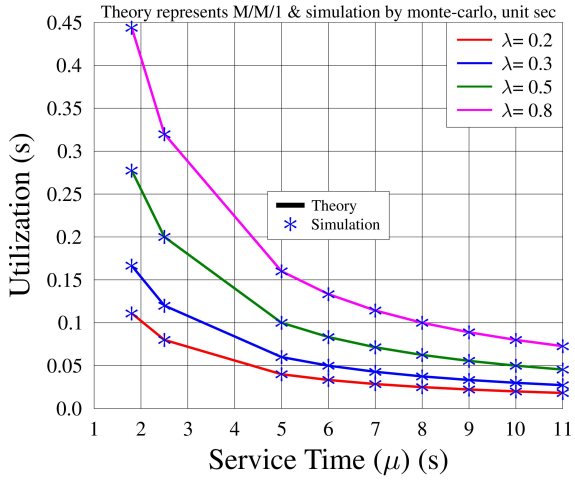

Fig. 4. Utilization Performance in Light of Service Time and Arrival Rate.

E. Average Utilization

Utilization performance, integrating both public and private blockchains, focuses on the fraction of time the server/miner is actively serving customers/creating blocks. It measures the workload of the server/miner. Fig. 4 reveals that utilization varies with λ and μ, impacting system performance. Higher λ relative to μ increases system utilization, indicating full capacity operation, while higher μ values with lower λ lead to underutilization.

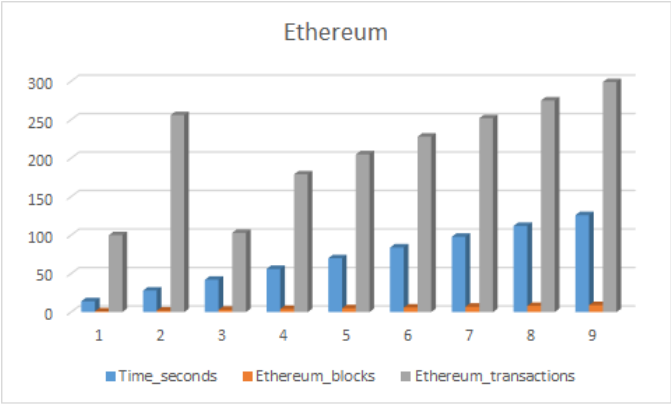

Fig. 5. Block Size of Ethereum Blockchain

F. Block Size of Hybrid Blockchain

Discussion on transactions and block size in hybrid blockchain environments highlights the efficiency of Ethereum and Hyperledger Fabric in processing transactions within

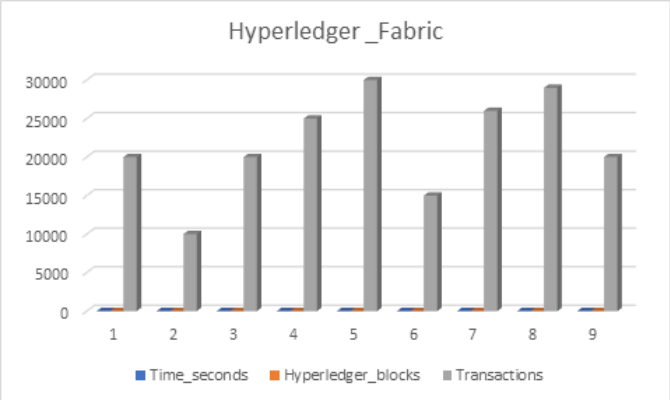

Fig. 6. Block Size of Hyperledger Fabric Blockchain

specific time frames. Similarly, Fig. 6 depicts Hyperledger Fabric's transaction handling and block generation capabilities, underscoring the platforms' transaction storage efficiency depending on miner activity.

G. Disease Detection using Machine Learning

Machine learning algorithms employed for disease prediction include MLP, SVM, SGD, and extra trees classifiers. Model evaluation follows, utilizing various metrics to assess

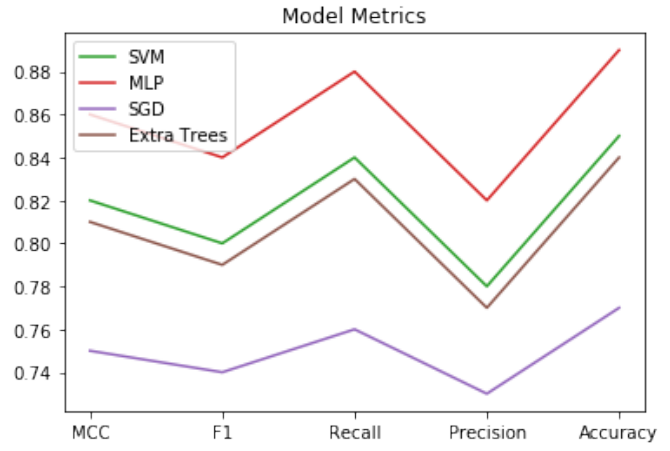

Fig. 7. Disease Detection using Machine Learning

performance. For SVM, the accuracy is 85%, precision is 0.78, recall is 0.84, F1 score is 0.80, and Matthews correlation coefficient (MCC) is 0.82. MLP shows an accuracy of 89%, precision of 0.82, recall of 0.88, F1 score of 0.84, and MCC of 0.86. SGD has an accuracy of 77%, precision of 0.73, recall of 0.76, F1 score of 0.44, and MCC of 0.77. Extra trees classifiers report an accuracy of 81%, precision of 0.79, recall of 0.83, F1 score

V. Conclusion

In the healthcare industry, it is common for different parties to share research data, medical information, and health records. Typically, this data is generated and transmitted by IoMT-based wearables, which need to be authenticated, and the data must be securely collected and transmitted. To address this, we present a hybrid method for the secure authentication of patient data using a combination of public and private blockchain technology. The proposed system, built on Hyperledger Fabric, offers a solution for trustless communication among enterprises in the healthcare sector. Additionally, we have incorporated access policies to limit user access. We also employed various machine learning algorithms for predicting diseases and storing the predictions on the blockchain for auditing purposes. For this feature, we are developing a MLP method, a SVM technique, a SGD strategy, and an ETC algorithm. In our system analysis, we use an M/M/1 queuing model and compare it to Monte Carlo simulation to highlight our system's performance in terms of low latency, higher throughput, and maximum utilization. The suggested healthcare blockchain architecture offers a potential solution to the industry's challenges. It provides a secure and transparent system that ensures the integrity of patient data while also allowing for secure patient data access and exchange. This research illustrates how blockchain technology can enhance the security and transparency of private medical data and opens new avenues for further studying IoMT-based healthcare systems.

References

[1] MarketsandMarkets. (2024) Healthcare it market by product (ehr, ris, pacs, vna, cpoe, hie, telehealth, healthcare analytics, population health management, supply chain management, crm, fraud management, claims management), end user (provider, payer) - global forecast to 2024. [Online]. Available: https://www.marketsandmarkets.com/Market-Reports/healthcare-it-252.html

[2] K. Fazlullah, A. Ryan, R. Md Arafatur, M. Spyridon, R. Imran, and S. Syed Tauhidullah, "Trustworthy and reliable deep-learning-based cyberattack detection in industrial iot," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 1.

[3] H. Journal. (2022, September) December 2021 healthcare data breach report. [Online]. Available: https://www.hipaajournal.com/december-2021-healthcare-data-breach-report/

[4] privacy igths.org. Data breaches. [(accessed on 08 november 2023)] available online. [Online]. Available: https://privacyrights.org/data-breaches

[5] F. Khan, M. A. Jan, R. Alturki, M. D. Alshehri, S. T. Shah, and A. ur Rehman, "A secure ensemble learning-based fog-cloud approach for cyberattack detection in iomt," *IEEE Transactions on Industrial Informatics*, 2023.

[6] B. Bhandari, R. Vairagade, H. Trivedi, H. Thakre, G. Indurkar, and A. Yadav, "Decentralized medical healthcare record management system using blockchain," in *2023 11th International Conference on Emerging Trends in Engineering & Technology-Signal and Information Processing (ICETET-SIP)*. IEEE, 2023, pp. 1–5.

[7] K. Baseer, B. J. N. Varma, B. Harish, E. Sravani, K. Y. Kumar, and K. Varshitha, "Design and implementation of electronic health records using ethereum blockchain," in *2023 Second International Conference on Electronics and Renewable Systems (ICEARS)*. IEEE, 2023, pp. 784–791.

[8] N. Iqbal, F. Jamil, S. Ahmad, and D. Kim, "A novel blockchain-based integrity and reliable veterinary clinic information management system using predictive analytics for provisioning of quality health services," *IEEE Access*, vol. 9, pp. 8069–8098, 2021.

[9] J. Zhou, C. Hu, J. Chi, J. Wu, M. Shen, and Q. Xuan, "Behavior-aware account de-anonymization on ethereum interaction graph," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 3433–3448, 2022.

[10] C. C. Agbo, Q. H. Mahmoud, and J. M. Eklund, "Blockchain technology in healthcare: a systematic review," in *Healthcare*, vol. 7, no. 2. MDPI, 2019, p. 56.

[11] S. Grampurohit and C. Sagarnal, "Disease prediction using machine learning algorithms," in *2020 International Conference for Emerging Technology (INCET)*. IEEE, 2020, pp. 1–7.

[12] M. J. Pattabhi and T. R. Konala, "Intent based access for policy control," in *2022 International Conference on Breakthrough in Heuristics And Reciprocation of Advanced Technologies (BHARAT)*. IEEE, 2022, pp. 43–48.
[13] V. C. Hu, D. Ferraiolo, R. Kuhn, A. R. Friedman, A. J. Lang, M. M. Cogdell, A. Schnitzer, K. Sandlin, R. Miller, K. Scarfone *et al.*, "Guide to attribute based access control (abac) definition and considerations (draft)," *NIST special publication*, vol. 800, no. 162, pp. 1–54, 2013.
[14] L. Karimi, M. Aldairi, J. Joshi, and M. Abdelhakim, "An automatic attribute-based access control policy extraction from access logs," *IEEE Transactions on Dependable and Secure Computing*, vol. 19, no. 4, pp. 2304–2317, 2021.
[15] Y. Wang, Y. Shen, C. Su, J. Ma, L. Liu, and X. Dong, "Cryptsqlite: Sqlite with high data security," *IEEE Transactions on Computers*, vol. 69, no. 5, pp. 666–678, 2019.
[16] T. Meng, Y. Zhao, K. Wolter, and C.-Z. Xu, "On consortium blockchain consistency: A queueing network model approach," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 6, pp. 1369–1382, 2021.
[17] J. E. Gentle, *Random Number Generation and Monte Carlo Methods*. Springer Science & Business Media, 2009.

Ateeq ur Rehman (Member, IEEE) is currently working as a Lecturer at the Department of Computing, Staffordshire University, UK. He received his Ph.D. degree from the University of Southampton, the UK in 2017. His area of research includes cyber security, blockchain, and privacy-preserved machine learning, particularly in healthcare and smart cities.

Nargis Tariq got an MS degree in Computer Science from the University of Haripur, Pakistan. Her research interest preserving privacy, Blockchain, and Deep Learning.

Dr. Mian Ahmad Jan Assistant Professor at the Department of Computer Science, School of Computing and Informatics, University of Sharjah, UAE. He received his PhD from the University of Technology Sydney, Australia. His research interests include cybersecurity, energy-efficient and secured communication for Internet of Things. He has been among the World's Top 2% scientists for the years 2021 and 2022.

Fazlullah Khan (Senior Member, IEEE) got PhD in Computer Science from Abdul Wali Khan University Mardan, Pakistan in 2020. He is an Assistant Professor at the University of Nottingham Ningbo China, Zhejiang, China. His research interests are Security and privacy, the Internet of Things, Machine Learning, and Health Informatics. He has been among the World's Top 2% scientists for the years 2022, 2023.

Houbing Herbert Song (M'12–SM'14-F'23) is currently a Professor at the University of Maryland, Baltimore County (UMBC), Baltimore, MD. He serves as an Associate Editor in IEEE Transactions and Journals and received multiple best papers awards. His research interests include cyber-physical systems/internet of things, cybersecurity and privacy, and AI/machine learning/big data analytics.

Muhammad Ibrahim got Ph.D. in computer science from the Capital University of Science and Technology, Islamabad, in 2019. He is working at the University of Haripur. His research interests include large-scale network simulation and modeling in blockchain cloud computing, VM migration, and task scheduling in cloud computing.