International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
***Article Received:** 14 August 2023 **Revised:** 04 October 2023 **Accepted:** 20 October 2023*

FLBP: A Federated Learning-enabled and Blockchain-supported Privacy-Preserving of Electronic Patient Records for the Internet of Medical Things

Satyanarayana Reddy Goluguri[1], Dr.Thammi Reddy Konala[2]
[1]Research Scholar, Department of Information Technology, GITAM University, Visakhapatnam. AP, India - 530045
Assistant Professor, Department of Information Technology, MVGR College of engineering, Vizianagaram, AP, India- 535005
E-mail: gsnreddy125@gmail.com
[2]Professor, Computer Science and Engineering, GITAM Institute of Technology, GITAM University, Rushikonda, Visakhapatnam, 530045, AP, India.
E-mail: tkonala@gitam.edu

Abstract—The evolution of the computing paradigms and the Internet of Medical Things (IoMT) have transfigured the healthcare sector with an alarming rise of privacy issues in healthcare records. The rapid growth of medical data leads to privacy and security concerns to protect the confidentiality and integrity of the data in the feature-loaded infrastructure and applications. Moreover, the sharing of medical records of a patient among hospitals rises security and interoperability issues. This article, therefore, proposes a Federated Learning-and-Blockchain-enabled framework to protect electronic medical records from unauthorized access using a deep learning technique called Artificial Neural Network (ANN) for a collaborative IoMT-Fog-Cloud environment. ANN is used to identify insiders and intruders. An Elliptical Curve Digital Signature (ECDS) algorithm is adopted to devise a secured Blockchain-based validation method. To process the anti-malicious propagation method, a Blockchain-based Health Record Sharing (BHRS) is implemented. In addition, an FL approach is integrated into Blockchain for scalable applications to form a global model without the need of sharing and storing the raw data in the Cloud. The proposed model is evident from the simulations that it improves the operational cost and communication (latency) overhead with a percentage of 85.2% and 62.76%, respectively. The results showcase the utility and efficacy of the proposed model

Keywords- ANN, Blockchain, Federated Learning, Elliptical Curve Digital Signature (ECDS), Blockchain-based Health Record Sharing (BHRS), Electronic Patient Records (EPRs), Internet of Medical Things (IoMT)

I. INTRODUCTION AND RELATED WORK

The Internet of Medical Things (IoMT) and the Cloud are the cutting-edge technologies being used to modernize and improve the healthcare sector. Process automation and data sharing are combined in the healthcare system to offer users individualized healthcare services and goods [1]. The process in order to run various healthcare applications and offer medical services to end users, IoT-based healthcare applications incorporate several industrial equipment or sensors. Additionally, these devices are utilized to gather patient body vital statistics, which are then saved as electronic patient records (EPRs) on cloud servers. The information is made available to doctors via on-demand queries in order to monitor patients' health conditions and administer therapy [2]. However, the centralization of healthcare providers and poor interoperability are problems that most existing healthcare systems must deal with. A patient would not, in particular, limit his or her options to a single hospital or physician. On the other hand, he or she can be transported between hospitals or visit several clinics or medical professionals for medical monitoring or therapy. This emphasizes the value of sharing patient EPRs among various medical facilities.

Due to the fact that they are accessed remotely, customized healthcare services can experience security and privacy problems. As a result, the patient's expectations were not met by the results of the ongoing intelligent treatment and remote monitoring. People require seamless mobility and resources to readily reach hospitals and high-quality healthcare [3, 4]. This Cloud-IoMT technology has the advantages of lower IT costs, reduced storage requirements, and increased productivity.

Massive amounts of medical data are being stored and made accessible over the Internet thanks in part to industrial cloud-based healthcare networks [5]. Personal privacy and data security issues in medical data systems continue to be crucial [6] because healthcare data is extremely dynamic and can travel from one site to another with various wireless connectivity. Furthermore, storing medical data on centralized third-party

Cloud servers increases security and privacy risks because the majority of the data is highly private and sensitive. In addition to causing financial or privacy breaches, these threats could lead to other attacks. Healthcare data is particularly vulnerable in the security framework and in the public domain, thus there is a great need to protect it. Additionally, with the existence of the physical gap between the IoMT and the Cloud, the communication and computation overhead significantly increases.

To safeguard the healthcare system from unauthorized users and attacks, a number of security measures are available [7], [8]. The standard encryption techniques used to safeguard the EPRs are unstable and violate user privacy because of the dynamic and open nature of the healthcare system. Prior to sending the EPR data to a cloud server, encryption is essential. Make sure the system is free of any invading users by checking as well. Numerous security and privacy concerns arise as a result of the decentralized storage and exchange of medical data. Although data-searchable encryption techniques have been established, a number of problems remain, including user-side and server-side verification and the storage, retrieval, and searching of medical data without the aid of a reliable third party. Additionally, there are some limitations to cloud computing in terms of security and privacy, the accuracy of patient data, service latency, and performance monitoring.

According to [10-12], the immutability, public verifiability, and programmability that blockchain technology possesses inherent benefits to address the aforementioned problems. Numerous studies [13, 9, 14] have recently used blockchain technology to distribute EPRs.

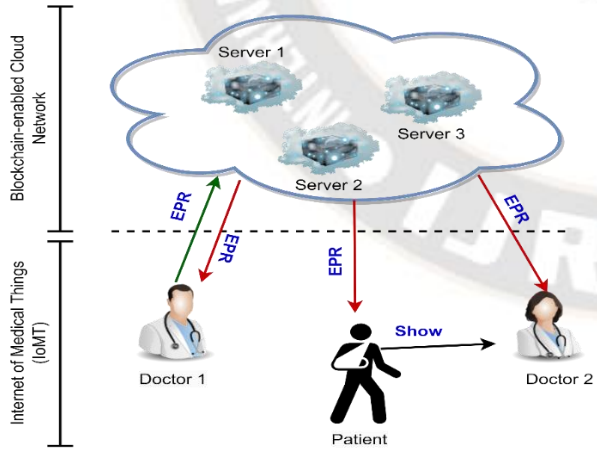
Blockchain-enabled Cloud Network
Server 1
Server 3
Server 2
EPR
EPR
EPR
EPR
Internet of Medical Things (IoMT)
Doctor 1
Show
Doctor 2
Patient

Figure 1. Traditional Blockchain-based EPRs Sharing (BEPRS) Framework

Figure 1 depicts a typical framework for blockchain-based electronic patient record sharing (BEPRS), where a doctor (or healthcare provider) is in charge of creating and releasing patients' EPRs into a blockchain-enabled Cloud network so that a patient can easily show their EPRs to other healthcare providers. These EPRs may optionally be stored locally or on distant trusted cloud servers to reduce the cost of blockchain storage, and just the summary information of EPRs (such as hash value, indexing, and timestamp) is recorded in a blockchain.

Patients and doctors can easily share and verify EPRs on existing BEPRS systems. However, sharing EPRs depends on the secured network architecture, and thus, interoperability and EPR privacy leakage will be a security issue. EPRs saved in blockchain are public to everyone, and malevolent spreading may lead EPRs to be disseminated after showing. We focus on the latter issue (EPRs can only be trusted by designated verifiers) because encryption and access control technologies can easily solve the former [15, 16]. For interoperability, the current study approaches an authentication mechanism for both users and service providers for the verification of data.

Digital signatures are required for EPR authenticity, non-repudiation, and integrity. This spreads harmful EPRs. EPRs and digital signatures can be sent to persuade others. Thus, the universal designated verifier signature proof (UDVSP, first proposed by Baek et al. [17]) may guarantee EPR verifiability, privacy, and anti-malicious propagation. Based on the UDVSP, a designator (e.g. patient) receives a new EPR with a signature from the signer (doctor-1). The patient can consult another doctor as Doctor-2 owns the patient’s EPR record.

To our knowledge, all UDVSP schemes (including [17, 18]) use bilinear pairing operations to guarantee BEPRS system features which are computationally complex and hinder its incorporation into BEPRS or other systems like electronic voting, anonymous certification, and income summary management [19]. An improved UDVSP method is implemented.

The contributions of this work are enlisted as follows:

- The proposed model makes use of an integrated Federated Learning-and-Blockchain-supported approach that has been formulated for the privacy-preserving of Electronic Patient Records (EPRs) in a collaborative framework;
- Through the classification of insiders and intruders using the given dataset, a secure classification model built on ANN has been created;
- For the interoperability issue, this work proposes a User-and-Service provider interaction authentication module;
- To reduce the communicational latency, an intermediate layer called the Fog layer is introduced as a complement to the IoT-Cloud framework to facilitate computations and improve security;
- An elliptic curve digital signature algorithm (ECDSA) is adopted to devise the bilinear pairing-free UDVSP scheme and frame a secured blockchain-based validation method;

- A blockchain-based electronic patient record sharing (BEPRS) is implemented to possess anti-malicious propagation;
- A Federated learning approach is integrated into the framework to form a global model without the need to share the raw data with the Cloud;
- In comparison to the method that is currently in use, the experimental study reveals that improvements have been made to classification accuracy, precision, recall, scalability, data privacy, and the identification of malicious conduct.

The rest of the paper is organized as follows: Section 2 studies the existing works based on UDVSP, ECDSA and the integrated FL-and-Blockchain privacy-preserving model. Section 3 presents the preliminaries of the techniques used in this work followed by the system framework. The proposed methodology is presented in Section 4. The performance evaluation along with the security analysis is elucidated in Section 5. Finally, Section 6 concludes the article with potential future directions.

II. REVIEW OF PREVIOUS STUDIES

Blockchain is one of the trending technologies available today, because it offers a platform that is both safe and dependable, making it ideal for the administration of data in a wide variety of applications, including the banking sector, the healthcare industry, and supply chain management. Additionally, the Internet of Things is starting to show promise as a potential use for blockchain technology. In this part, various research papers that were published in the not-too-distant past are discussed in this section in order to investigate and have a better understanding of the roadmap of its function in connection with IoT.

In order to securely retain healthcare certificates, Sharma et. al. [20] proposed a methodology for the distributed application's privacy protection. The ether scan tool was used to conduct an assortment of trials to quantify the operation cost, latency, and processing time in order to assess the performance of the suggested model. Lin et al. [21] have proposed the EMR chain model, which combines a bilinear pairing model that is compatible with blockchain technology and also uses a UDVSP scheme that takes no time at all, to address the long-standing problems with accessing electronic medical records in a centralized environment. In order to make the suggested system an anti-malicious propagation system, the UDVSP scheme employs the elliptical curve as the name of the digital signature method. For the purpose of maintaining the confidentiality of electronic health records (EHR), Alzubi et al. [22] have presented a hybrid model that is a blend of the deep learning model with the technology behind blockchains. The model that was presented has improved performance but at the expense of an increased amount of time consumption. Addressing the concerns of data privacy and security in a smart city's Internet of Things setting, Singh et. al. [23] proposed a solution which is a Blockchain and Federated Learning-enabled Secure Architecture for Privacy-Preserving in Smart Healthcare. Privacy is maintained, scalability is achieved, and data sharing is optimised through the integration of Blockchain-based IoT cloud platforms and Federated Learning technologies. Tackling data privacy problems and diverse model architectures in Healthcare 4.0, Veronika Stephanie et. al. [24] presented a Secure Multiparty Computation-based Ensemble Federated Learning with a Blockchain solution. With the suggested architecture, hospitals and other medical facilities may work together to enhance the global model while also creating their own model structures. In this study [25], the authors investigate whether or not it is possible to improve the field of evidence-based medicine by merging machine learning with blockchain technology and privacy-preserving encryption approaches. In order to do predictive analysis in individualized healthcare utilizing EHRs, Gupta et al. [26] presented a unique supervised Deep Similarity Learning technique. In order to build patient representations and capture interactions between patients, the suggested technique uses CNN-Softmax, a Siamese-based neural network that leverages pairwise similarity learning. The model outperforms conventional similarity learning techniques by conducting illness prediction with outstanding accuracy (97.8%) using Convolutional Neural Networks (CNN) and Softmax-based supervised classification. Zaman et. al. [27] presented a study that introduces a revolutionary Holochain-based architecture for protecting the privacy and security of IoT healthcare systems. In contrast to the blockchain, Holochain is inherently decentralized and user-centric since apps run locally on each user's device, and is ideal for instances where resources are limited since it solves the scalability problem presented by blockchain. The findings point to the possibility of the widespread implementation of IoT healthcare systems that are both efficient and protective of patient privacy. In this study [28], the authors present a regular pattern mining model for personalized healthcare utilizing IoT data that is built on a convolutional neural network (CNN). To reliably forecast abnormal health problems from unstructured medical health information, the suggested approach makes use of the Pearson Correlation Coefficient and regular pattern behavior. Significant health variables are identified, categorized using correlation analysis, and regular patterns connected to obesity, hypertension, and diabetes are discovered using this approach. Jia et. al. [29] presented a study that provides a federated learning data protection aggregation strategy for the IIoT that makes use of blockchain technology. This study provides essential assistance for data security in IIoT businesses and adds to improving safe data transmission and sharing in industrial environments. The work by Li, D., et al. [30] provides a comprehensive review of Blockchain-federated learning (BCFL) as a potential decentralized deep learning system. BCFL combines Blockchain's security and speed improvements with federated learning's privacy-preserving strengths. In order to address privacy and efficiency concerns in fog computing, Qu et. al. [31] introduced FL-Block, a revolutionary blockchain-enabled federated learning method. FL-Block allows for the decentralized privacy protection and poisoning attack resistance of a blockchain-based global learning model to be swapped with local learning updates of end devices. To improve privacy and security in IoHT applications, Rahman et. al. [32] present a lightweight hybrid federated learning (FL) architecture based on blockchain-based smart contracts. Xu, J., et al. [33] presented a study that presents FNCF, a privacy-protecting, individually-tailored blockchain reliability prediction model based on federated learning neural collaborative filtering for Internet of Things settings. The approach protects users' privacy by not sharing critical context information with other parties. Offloading and scheduling difficulties in healthcare processes inside the IoMT fog-cloud network are investigated in a study published by Lakhan, A., et. al. [34]. A unique deep reinforcement learning and blockchain-enabled system is suggested, which combines task sequencing and research matching techniques with blockchain task scheduling.

In the realm of healthcare and IoT applications, the integration of blockchain and federated learning technologies has ushered in a promising era of data privacy and security. However, striking the delicate balance between data privacy and utility, while guarding against re-identification risks, requires further research and development. Standardization and regulatory compliance must be addressed to enable seamless data sharing and ensure adherence to stringent healthcare data regulations. Moreover, the real-world applicability and resource requirements of these innovative approaches call for rigorous testing and validation in practical healthcare settings. Combining creativity and a meticulous approach will propel these technologies towards their full potential, revolutionizing the healthcare landscape and empowering patients with secure, privacy-preserving IoT-driven medical solutions.

III. PROPOSED METHODS

This section proposes a system framework followed by the techniques used in this work such as Artificial Neural Network (ANN), Elliptical Curve Digital Signature Algorithm (ECDSA), Universal Designated Verifier Signature Proof (UDVSP), Federated Learning (FL), and Smart contract.

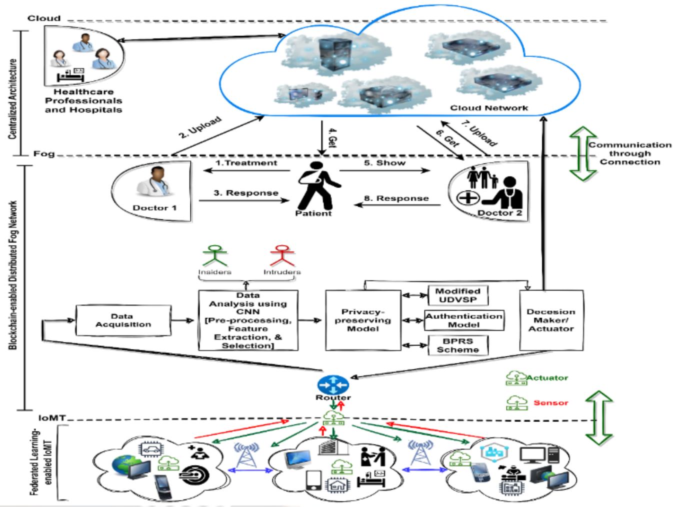

Figure 2. A Blockchain-based Electronic Patient Record Sharing system in a collaborative IoMT-Fog-Cloud environment

A. *System Framework*

Figure 2 showcases the Blockchain-based Electronic Patient Record Sharing (BEPRS) system in a collaborative IoMT-Fog-Cloud architecture. This architecture consists of three layers such as (1) Internet of Medical of Things (IoMT) as the ground layer, (2) Fog layer as the intermediate and complementary layer, and (3) Cloud layer as top layer.

The ground layer is an IoMT layer which consists of several internet-enabled medical things, such as smart medical equipment, smart gadgets, smart devices, etc. with different specifications. All these devices are federated learning (FL)-enabled devices which enable security features in the proposed collaborative architecture. All these devices produce voluminous data which are sensed through sensors and forwarded to the computing nodes through networking devices. In addition, these devices are used to access the patient's records through Cloud servers. Healthcare experts and patients work in this layer to access the EPRs from the Cloud server through Blockchain network. Next, a complementary to the Cloud layer called the Fog layer is introduced in this architecture to provide seamless computation on the edge of the network. In the existing literature, an integration of IoMT and Cloud layer is exhibited for sharing and storing the EPRs. However, the physical gap that exists between the IoMT and the Cloud servers results in increasing the latency and communication overhead. Moreover, the unawareness of the physical location of the data in the Cloud datacenter leads to security and privacy issues. The introduction of the Fog layer between the Cloud server and the IoMT not only reduces the incurred latency but also brings the computations nearer to the end devices over the edge of the network. Furthermore, the centralized storage and centralized architecture of the Cloud are now surmounted by the distributed nature of Fog computing. This layer, in this architecture, encompasses heterogeneous resources for computation along with a privacy-preserving model that facilitates the identification of insiders and intruders users through ANN, building a secured BEPRS scheme through a proposed model, and a validation scheme through ECDSA. This layer contains many components responsible for generating and accessing the EPRs, such as Doctors, Patients, and Blockchain network. Doctors: This entity is in charge of keeping patients' EPRs and offering diagnostic services. The doctor creates an EPR after receiving a patient's outpatient care request (Step 1) and then conceals it (with the EPR signature) with blinding variables. Then, in step two (for later retrievals), the doctor uploads the blinding EPR signature onto the blockchain network. In addition, the patient will get the blinding elements as a "Response" (Step 3). Additionally, while providing diagnostic services, this entity may need the patient to submit historical EPRs (Step 5) and obtain relevant data from the blockchain for verification (Step 6). The alternate doctor also uploads the updated EPR with diagnostic services into the Blockchain (Step 7). Patients: Owner of the EPRs, this entity acquires fresh EPRs and blinding factors following diagnoses and displays his or her prior EPRs before diagnosis. The authenticity of blinding EPRs linked on blockchain may be verified through blinding factors (Step 4). When a patient visits a different doctor, the patient can submit historical EPRs to get more precise and effective medical care. However, to prevent the harmful spread of EPRs, the patient uses his or her blinding characteristics to demonstrate ownership of one EPR without explicitly displaying the EPR (Step 5). Blockchain network: This entity is responsible for keeping blinding EPRs up to date and offering uploading (Step 2 & Step 7), checking (Step 4), and getting (Step 6) services linked to EPRs. It could be joined if it's public, or else it might be permissioned. Unlike the former, which is exclusively kept up by members with permission, the latter kind is maintained by anybody. Any blockchain that supports smart contracts might be used in our proposal. At the top, the Cloud layer consists of many servers used to store and access the EPRs on demand from anywhere. In our architecture, medical professionals and patients are allowed to access the Cloud servers through a distributive model for accessing EPRs through an authentication mechanism.

Besides, the above system architecture ought to satisfy the following properties. (1) Compatibility: this property ensures the compatibility and interoperability of several devices supporting blockchain i.e., public blockchain and private blockchain. (2) Completeness: this property ensures the acceptance of proof of owning the EPRs with an authorized blinding factor. The EPRs cannot be generated without an authorized blinding factor. (3) Anti-malicious propagation: this property ensures preventing the EPRs from being manipulated or propagating maliciously. For instance, if Doctor _1 shares the patient’s EPR with another Doctor_2 then the Doctor_2 must trust the suggestions given on the EPR by Doctor_1 and should be propagated malevolently. (4) Unlinkability: this property ensures that the two EPRs with blinding factors cannot be merged into one to preserve the privacy of EPRs.

B. Blockchain

A blockchain is a form of public ledger or distributed database where verified transactions and digital events are saved and chronologically connected in data blocks. The records created by the transaction verifier and the data provided by the transaction initiator together make up the so-called data block. Additionally, each block has a timestamp and the hash of the block before it, making the data in the blockchain unchangeable and traceable. Valid blocks will be uploaded to the blockchain whenever there is consensus among 51% of the distributed network's users. This distributed P2P network is also robust against single-point failure and attacks because each node reserves the same copy of transaction records. Consequently, blockchain has received a lot of attention in a variety of industries.

C. Artificial Neural Network (ANN) for malicious user detection system

The ANN structure mimics the interconnected neurons found in the human brain. The Input layer, the Hidden layer(s), and the Output layer are some examples of the three layers that make up its architecture. The input layer receives the input data, while the hidden layer or layers between the input and output layers represent a large number of neurons. In an ANN, each layer processes information from the Input layer in the Hidden layer and outputs them in the Output layer. The weights of all the preceding nodes are used to calculate the output of each node using an activation function. An Adam optimizer is employed in the hidden layer to train the network more quickly and effectively. Figure 3 displays the suggested ATDS framework. To manage the coordination of all the nodes, this is implemented into the server at the Fog layer. This framework is made up of important components including Data preparation, Data Augmentation, and a Fully linked layer. The data acquisition module of the data pre-processing gathers the data from several servers and pre-processes it. The Data Augmentation module then divided the entire data set into the Training set and the Validation set. The dataset is further subjected to feature extraction and reduction in order to recover the pertinent intrusion-related features, and classification is then performed on the training dataset to give the ANN instructions on how to perform as accurately and quickly as possible. The output module then generates the detection rate pertaining to the threat or non-threat. A UNSW-NB15 dataset has been used for the testing phase [35].

***Article Received:** 14 August 2023 **Revised:** 04 October 2023 **Accepted:** 20 October 2023*

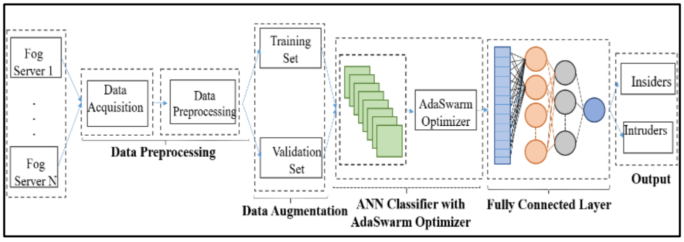

Figure 3. ANN-based Malicious Users identification

The ANN is a widely used mathematical framework used for information processing that models the information transmission process of the human brain through neurons. The input, hidden, and output layers are the three fundamental layers of the ANN, which facilitate the prediction process. The acquired dataset's traits or qualities comprise the input layer, which the intelligent system receives. The hidden layers do the necessary calculations by applying a non-linear modification to the input data. After receiving information from the hidden layer, the output layer then generates the outcome. The Multi-Layer Perceptron (MLP) is taken into consideration in this study. The MLP is a feed-forward neural network enhancement in which the signal only flows forward. With the exception of the input nodes, each node in an MLP computes a weighted sum of the nodes from the layer preceding it. Mathematically, it can be stated as follows:

$$\theta_t = \left[\sum_{i=1}^{n} \omega_{ij} \alpha(\ln(i))\right] + \varphi_k \qquad (1)$$

Where the bias for the output layer is represented by φ_k. Assuming that $i = 1,2,\cdots,n$ and $j = 1,2,\cdots,m$, denote the nodes in the hidden and output layers, respectively, ij signifies the weight between the nodes of the hidden layer and output layer. In this instance, the activation function is represented by,

$$\ln(i) = \left(\sum_{l=1}^{s} \rho_l \omega_{li}\right) + \varphi_i \qquad (2)$$

Where ρ_l stands for the number that corresponds to the l^{th} node in the input layer with the value $l = 1,2,\cdots,s$.

As a result, we use the sigmoid function as the activation function to process the result of Eq. 2 which can be denoted through the following representation,

$$sigmoid(\rho) = \frac{1}{(1+e^{\rho})} \qquad (3)$$

Now, by using the values in Equations (1) and (2) in Eq. (3), we obtain,

$$\theta_t = \left[\sum_{i=1}^{n} \omega_{ij} \alpha\left(\left(\sum_{l=1}^{s} \rho_l \omega_{li}\right) + \varphi_i\right)\right] + \varphi_k \qquad (4)$$

D. Elliptic Curve Digital Signature Algorithm (ECDSA)

This scheme is used to form the bilinear pairing for the UDVSP. This algorithm involves four procedures, namely, *Setup, Electronic Key Generation (EKG), Electronic Signing (ESign), and Electronic verification (EVer).* All these procedures are explained as follows:

- *Setup:* this step is represented $p \leftarrow Setup(1^{\delta})$ and a combination of input and output. This step takes input, namely, δ as a security parameter and produces output p as a public parameter, where $p = \{E_c, F_p, S, I, G, r_1, r_2, H_f\}$. r_1 and r_2 are primer numbers of δ bits, E_c is an elliptical curve defined by $a^2 = b^3 + xb + y\%r_1$ $(x, y \epsilon F_p)$, S is an additive set consisting of all the parameters of E_c, and an infinity point I, F_p is a finite state consisting of p elements, G denotes the generator of S of order r_2, and H_f denotes the hash function expressed as $Z_{r_2} \rightarrow \{0,1\}^*$.
- *EKG:* this step is computed through input as a public parameter p and produced output in terms of public and secret keys. It arbitrarily selects a secret key s_k $\left(d_u(\epsilon Z_{r_2})\right)$ to compute the public key P_k $\left(P_u(= d_u G)\right)$. This can be expressed as $EKG(p) \rightarrow (d_u, P_u)$.
- *ESign:* this step takes the system public parameter p and a secret key $s_k(= p, d_u)$, and a message as inputs. It arbitrarily selects $k \epsilon Z_{r_2}$ to estimate $K = k_G = (a_k, b_k)\%r_2$. It returns an output message in the form of $\alpha(r,s)$, where $r = a_k\%r_2, s = k^{-1}\left(H_f(message) + d_u r\right)\%r_2$. It is expressed as $ESign(p, s_k, message) \rightarrow \alpha(r,s)$.
- *EVar:* this step of the algorithm takes the system public parameter p, public key $P_k = (p, P)$, a message and a $ESign\left(\alpha(r,s)\right)$. It scans $ESign$ and estimates $K' = \left[s^{-1} H_f(message)\right]G + (s^{-1}r)P = (a'_K, b'_K)$, and $r' = a_{K'}\%r_2$. If $r' = r\%r_2$ then it returns 1 (as an indication of valid), otherwise, returns 0. It is expressed as $EVar\left(p, P_k, message, \alpha(r,s)\right) \rightarrow \{0,1\}$.

E. Universal Designated Verifier Signature Proof (UDVSP)

This UDVSP is consisting of six sets of procedures, namely, *Setup*, *KGen*, *ESign*, *Verf*, *Transform*, and *IVer* [20]. Each procedure is explained as follows:

- *Setup:* this step of the algorithm includes two parameters as input and output, namely, security parameter (δ) and a system public parameter (p). It is expressed as $p \leftarrow Setup(1^{\delta})$.
- *KGen:* this step of the algorithm includes an input as system public parameter p, and an output as a combination of a public key P_k and a secret key s_k. It is expressed as follows: $EKG(p) \rightarrow (s_k + P_k)$.

***Article Received:** 14 August 2023 **Revised:** 04 October 2023 **Accepted:** 20 October 2023*

- *ESign:* this step takes the system public parameter p and a secret key s_k, and a message as inputs. It returns an output message in the form of $\alpha(r,s)$. It is expressed as $ESign(p, s_k, message) \rightarrow \alpha(r,s)$.
- *EVar:* this step of the algorithm takes the system public parameter p, public key P_k, a message and a $ESign(\alpha(r,s))$. It returns 1 as an indication of a valid signature of the message, otherwise, returns 0. It is expressed as $EVar(p, P_k, message, \alpha(r,s)) \rightarrow \{0,1\}$.
- *Transform:* this step of the algorithm includes public parameter p, public key P_k and a valid signature $\alpha(r,s)$ as inputs and produces a pair of transformed signature and a secret key. It is expressed as $Transform(p, P_k, \alpha(r,s)) \rightarrow (\alpha'(r,s) + s'_k)$.
- *IVar:* this verification is run between a designator (P) and a designated verifier (V). This step takes a public parameter p, a public key P_k, a message, and an updated signature $(\alpha'(r,s))$ as inputs. The input of the designator is the secret key (s'_k) and for the designated verifier, it is null. The key objective of the designator (P) is to verify the transformed signature from the original signature $(\alpha(r,s))$. Based on the verification, it returns 1 if it is validated & verified, otherwise, returns 0. It is expressed as follows: $IVar[P(s'_k) \leftrightarrow V](P, P_k, message, \alpha'(r,s)) \rightarrow \{0,1\}$.

Each entity in this algorithm has a specified role. For instance, the steps $KGen$ and $ESign$ have been executed by the signer and the designator performs verification $(EVar)$ and $Transform$ operations. Moreover, there are two communications that take place in this scheme. First, it is between the $ESign$ and $EVar$, i.e., the signature generated by the $ESign$ should be accepted by the $EVar$ step. The second is between $IVar$ and $Transform$, that is, the transformed signature with specified inputs should be accepted by the $IVar$ step of the algorithm.

F. *Federated Learning (FL)*

Traditional ML techniques gather and process data on a central server. As a result, the computing and communication overheads on the central server increase. Because ML's usefulness and accuracy depend on the amount of data and the server's capacity, this raises challenges. The training data may also disclose sensitive information about the data and its owner. As a result, the owner of the data is hesitant to make it public, especially when it comes to medical data. The incapacity of the central server to safeguard the data, which precludes the data owner from sharing their medical history for training, is another significant issue. As a result, privacy concerns prevent the data from being used effectively. Google created a brand-new concept known as "federated learning" in an effort to allay data owners' privacy worries. It enables the data owners to collectively upload their data into a global model without releasing the actual data [36]. This learning model can be separated into two groups from the perspective of networking: (a) Centralized FL and (b) Decentralized FL. In the centralized FL model, the local models are brought together via a central server into a single global model. In the decentralized FL, however, the aggregating process is carried out by each data owner. The fundamental FL model is trained using patient-specific data in our process. The FL operates in this manner. Consider a central server with a FL job, N trainers, and the server. While the central server has an initial global model, each trainer has a local model. The server first sends the first global model to each trainer. All of the trainers train the global model using their local data after it has been received, and they then update their local models before sending them to the central server, which retains the original data. The global model is then updated as necessary by the central server, which then merges all of the local models. Using the local data that was supplied to each trainer by the central server, the trainers subsequently update the overall model. Until the global model converges or has gone through the maximum number of training iterations, this process is repeated. In order to safeguard the privacy of the raw data and the data owner, privacy and sensitive information are stored locally because the trainers must submit changes to the models. Since trainers upload only the alterations into the global model rather than the original data, FL minimizes communication and computation overheads due to its effective high communication capabilities. Figure 4 shows the FL-enabled privacy-preserving model.

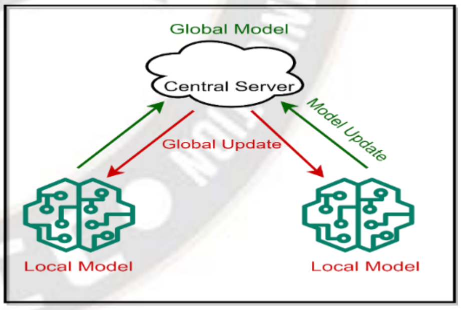

Figure 4. Federated Learning (FL)-enabled Privacy-preserving model

G. *Smart Contract*

On the public ledger, smart contracts are event-driven computer programs. It is capable of managing and transferring assets with high value. The open source blockchain network Ethereum is a well-known use case for smart contracts. Smart contracts are specifically scripts or codes that are used in blockchain. The scripts on the contract content could be run without the aid of an external trusted authority once the predefined circumstances have been activated. The whole procedure is computerized, and the completed transactions are logged on the public ledger for auditing. The owner of the asset has the authority to cancel the user's access privileges if they break the terms of the agreement. To precisely govern the data sharing of EPRs, the proposed method allows patients to predefine access rights, access actions (read, write, or copy), and access duration in smart contracts.

To manage blinding EPR signatures (including uploading, checking, obtaining, and revoking), our EPRChain system uses smart contracts. Only if it was entered into the smart contract by authorized doctors is one blinding EPR signature valid. Patients can then adopt these obscuring elements to produce reliable proof. Others can obtain these blinded EPR signatures to confirm the legitimacy of the patients' documentation. Additionally, blinding EPR signatures can only be uploaded or revoked by approved doctors. Thus, the four algorithms Upload, Check, Get, and Revoke are the essential components of our developed smart contract.

IV. PROPOSED METHODOLOGY

This section presents the modified UDVSP and EPR system along with an EASEID (Efficient Access management and Session-based Electronic Identification Distributed) model.

A. *Modified UDVSP Scheme*

The ECDSA is incorporated in the UDVSP to avoid time-consuming issues in bilinear pairing in UDVSA. The detailed scheme is illustrated here:

- *Setup:* this step is represented $p \leftarrow Setup(1^{\delta})$ and a combination of input and output. This step takes input, namely, δ as a security parameter and produces output p as a public parameter, where $p = \{E_c, F_p, S, I, G, r_1, r_2, H_f\}$. r_1 and r_2 are primer numbers of δ bits, E_c is an elliptical curve defined by $a^2 = b^3 + xb + y\%r_1 \, (x, y \epsilon F_p), S$ is an additive set consisting of all the parameters of E_c, and an infinity point I, F_p is a finite state consisting of p elements, G denotes the generator of S of order r_2, and H_f denotes the hash function expressed as $Z_{r_2} \rightarrow \{0,1\}^*$.
- *EKG:* this step is computed through input as a public parameter p and produced output in terms of public and secret keys. It arbitrarily selects a secret key $s_k \left(d\left(\epsilon Z_{r_2}\right)\right)$ to compute the public key $P_k \left(P(= d_u G)\right)$. This can be expressed as $EKG(p) \rightarrow (d, P)$.
- *ESign:* this step takes the system public parameter p and a secret key $s_k(= p, d)$, and a message as inputs. It arbitrarily selects $k \, \epsilon \, Z_{r_2}$ to estimate $K = k_G = (a_k, \ b_k)\%r_2$. It returns an output message in the form of $\alpha(r, s)$, where $r = a_k\%r_2, s = k^{-1}\left(H_f(message) + dr\right)\%r_2$. It is expressed as $ESign(p, s_k, message) \rightarrow \alpha(r, s)$.
- *EVar:* this step of the algorithm takes the system public parameter p, public key $P_k = (p, P)$, a message and a $ESign\left(\alpha(r, s)\right)$. It scans $ESign$ and estimates $K' = \left[s^{-1} \, H_f(message)\right]G + rP = (a'_K, b'_K)$, and $r' = a_{K'}\%r_2$. If $r' = r\%r_2$ then it returns 1 (as an indication of valid), otherwise, returns 0. It is expressed as $EVar\left(p, P_k, message, \alpha(r, s)\right) \rightarrow \{0,1\}$.
- *Transform:* given a public parameter p, the public key P_k, and a valid signature $\alpha = (r, s)$, this step arbitrarily selects $x, y \epsilon Z_{r_2}$ to estimate $r' = r + x\%r_2$ and $s' = s + y\%r_2$. It returns the transformed signature as $\alpha' = (r', s')$.
- *IVar:* there are four operations that take place between P and V:

(a) First, P estimates $K' = \left[s^{-1} \, H_f(message)\right]G + rP$, and it randomly selects $\tau, \beta \, \epsilon \, Z_{r_2}, R_1, R_2 \, \epsilon \, G$ to estimate $K' = K + R, R' = xR, D = s'R_2 + \beta K' - \tau P$. Afterwards, P send (K', R', D) to V.

(b) Second, V arbitrarily selects a challenge $c \, \epsilon \, Z_{r_2}$ and sends it to P.

(c) Third, P estimates $Z_R = R_2 - cR_1, Z_a = \tau - ac\%r_2$ and $Z_b = \beta - bc\%r_2$, and it forwards (Z_R, Z_a, Z_b) to V.

(d) Fourth, V estimates $D' = s'Z_R + Z_b K' - Z_a P + c[s'K' + R' - r'P - H'(message)G]$ and verifies that $D' = D$ is true or not. It returns 1 if it holds true, otherwise, it returns 0.

B. *Modified EPRChain Framework*

In this subsection, we explain our developed EPRChain system based on the aforementioned UDVSP scheme UDVSP = (KGen, ESign, EVer, Transform, IVer) and smart contract (Upload, Check, Get, and Revoke).

- *Setup:* in this initialization phase, the Manager calls *Setup* to produce a public parameter p , and deploys it in blockchain with a smart contract to get a smart contract id SID for contract identity. The system manager then shares (p, SID) to other parties (such as patients and doctors) after which utilising cryptography algorithms to start the smart contract. Additionally, every doctor activates $KGen$ to create a secret key $(s_k = d)$ as well as a public key $(P_k = P)$.
- *Signing up (Registration):* in order to receive the authorised blinding EPR signatures, patients must go through this registration procedure. The doctor specifically employs a secret key (s_k) which determines the patient's original EPR signature by using $\alpha = ESign(s_k, msg)$, where msg is the diagnostic code for the patient message. After that, the physician uses $Transform$ procedure to create the blinding EPR signature of α and blinding Secret keys (a, b), i.e., $(\ \alpha', a, b\ ) = Transform(\alpha)$. Next, to upload the transformed signature α', the doctor invokes the $Upload$ in SID through the smart contract. At last, the blinding secret

***Article Received:** 14 August 2023 **Revised:** 04 October 2023 **Accepted:** 20 October 2023*

key (a, b) with the retrieving index $index = H_f(\alpha')$ is forwarded to the patients over a secured network. Correspondingly, the patient calls $Get(index)$ method to get the blinding information on the transformed signature. In addition, the patient uses the blinding technique to get the original secret code α to reassemble the authentic EPR signature, and calls $IVer$ to verify that is valid.

- *Show.* A patient will use this stage to privately show other doctors (like Doc_2) their EMR signature. The patient does not have to send his/her EPR, and then Doc_2 will accept the patient's signature but will believe it is her own signature here. In particular, the patient and Doc_2 carry out the interactive protocol $IVar$ in this sentence. To get the patient's blinding EPR signature, Doc_2 must call Get method to extract it from SID. If the $IVar$ returns 1, EPR will belong to the patient and Doc_2 believes that the signature belongs to the patient; otherwise, Doc_2 will Refuse to believe this assertion.
- *Cancellation.* The doctor starts this step to revoke a blinding EPR signature α', i.e., the doctor initiates $Revoke$ method to delete the α' to delete it from the smart contract. Simultaneously, it can be checked if it has been revoked or not by others by calling $Check$ function.

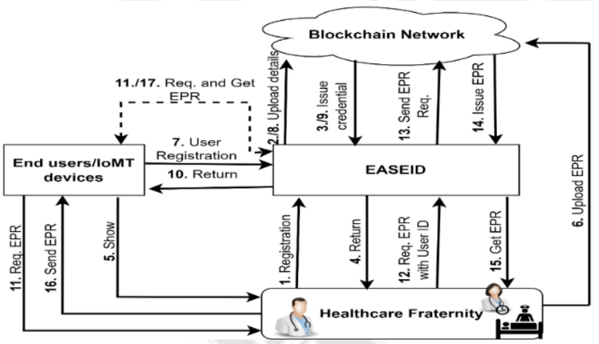

Figure 5. EASEID model

C. *Efficient Access Management and Session-based Electronic Identification and Distributed (EASEID) Model*

Figure 5 shows the efficient access management and session-based electronic identification and distributed model. This model showcases the interaction of users (end users/IoMT devices) and healthcare fraternities to the proposed EASEID model in conjunction with a blockchain network. This model is an interface between end-users/healthcare professionals and a blockchain network depicting the interaction and communication for requesting/uploading/getting EPRs.

The production and maintenance of official health documents employing a variety of phases, including the acquisition, representation, validation, and justification of healthcare data, is proposed in this research using blockchain-based EASEID. Various IoMT devices are used by users to administer medical certificates. There are no precise requirements for the device the user uses to retain the certificate. The EASEID offers a user interface so that users can generate, save, and validate medical certificates, among other services. It uses a distributed blockchain network to construct a web application. The user first registers in the EASEID and is given the special ID by the relevant authorities during registration. The healthcare authorities first confirm the users' records as healthcare experts if any user or patient contacts the healthcare centres to get an official health document. The necessary medical certificate is then generated via the blockchain network. To create blockchain-based documents with a distinct ID, the EASEID then processes or deploys the certificate on the blockchain network. The healthcare document is then saved as a transaction with a distinct block-based blockchain ID. The backend of the proposed IoMT-based architecture, which is a distributed file system, supports the public blockchain. There is a user interface to interact with the suggested system at the front end. The EASEID creates medical certificate records using the Ethereum public network. The consensus algorithm used by the proposed architecture is called Proof of Work (PoW). The distributed ledger is maintained by the EASEID, which also offers protection against unauthorised insertion, deletion, and updating of medical records.

The distributed application based on a blockchain system for protecting medical records is shown in Figure 5. Four components make up the proposed system: users/IoMT devices, healthcare specialists, EASEID, and a blockchain network. The distributed application receives an initial registration request from hospitals or physicians of healthcare facilities. Following that, the EASEID verifies registration data, gets the entity's credentials from the Ethereum network, and stores the entity's data on the blockchain network. Similar to this, IoMT devices like wearable smartwatches and many others register in the proposed architecture by submitting the request to the EASEID with the unique identification number and carrying out the aforementioned registration process. The EASEID then presents the medical professionals with the gathered credentials. The user of the suggested application then sends a request for the medical records from the registered medical professionals in the system. Now, the medical professionals transmit the appropriate request to the EASEID together with the user information. Following request submission, the EASEID creates the desired medical document with a special ID and keeps user information in the suggested network. The user and the medical facility are then given access to the certificate's special ID by the EASEID. Finally, the user can utilise the special shared ID to access the created medical document. Additionally, by connecting the registered IoMT device to the suggested architecture via Wi-Fi, the user can obtain the created certificate in the IoMT device. The suggested system is as follows successive actions:

Step 1: Healthcare facilities send the distributed application a registration request together with the username, address, unique ID, etc.

Step 2: After receiving the registration information, the EASEID confirms that the healthcare professional is already registered with the EASEID database. The EASEID stores the information about the experts in the blockchain after the verification network.

Step 3: The blockchain network generates the login information after saving the information of the medical professionals and the distributed application receives the healthcare expert's credentials.

Step 4: The EASEID provides the healthcare expert with the credentials and grants the expert access to the architecture services that are suggested.

Step 5: Users show their health to Doc_1.

Step 6: The healthcare professionals upload the patient's EPR into the blockchain.

Step 7: Users send the EASEID a registration request with the username, address, unique ID (if it has been issued), etc.

Step 8: After receiving the registration information, the EASEID confirms that the healthcare professional is already registered with the EASEID database. The EASEID stores the information about the experts in the blockchain after the verification network.

Step 9: The blockchain network generates the login information after saving the information of the medical professionals and the distributed application receives the healthcare expert's credentials.

Step 10: The user receives the desired credential from the distributed application.

Step 11: Users ask for the EPRs from healthcare experts. The dashed lines (Step 11 & Step 17) indicate that the user can also fetch its EPR from the EASEID upon signing up with credentials.

Step 12: As medical professionals, healthcare facilities verify the accuracy of the information they have received from the user. For instance, a healthcare professional checks a person's health status and severity, length of the treatment, hospital consultant doctor, etc. if the user requests a sick certificate. Then, it makes a decision in accordance with the user's request, whether or not to move further.

Step 13: Medical facilities process user requests in this phase to confirm the credentials and to create a medical certificate, then connect to the blockchain via the EASEID and Metamask wallet request.

Step 14: The blockchain network stores the user information along with the information for the created certificate in the blockchain architecture and provides the EASEID with the produced unique certificate ID.

Step 15: The EASEID provides the user and the healthcare provider with the generated certificate ID.

Step 16: the healthcare experts issue the required EPR to the patient.

Step 17: Using the EASEID, the user is given access to the medical certificate that was generated by simply using the specific ID provided by the certificate.

V. PERFORMANCE EVALUATION

In this section, the experimental setting is described, the performance metrics are assessed, and a comparison of the suggested work is made.

A. Simulation Environment

The Ethereum platform is used to implement the suggested work. Smart contracts are implemented on the open-source Ethereum public blockchain network. The suggested distributed application's many functions are provided by the smart contract. Users can sign up, create, check the validity of, and access medical certificates using this tool. Additionally, the smart contract detects illegal manipulation and unauthorised access and guards against assaults on the suggested design. The testing environment for the proposed work includes the Testnet for executing tests on the proposed work based on various performance evaluation parameters, including latency, processing time, throughput, and response time, as well as the Ganache tool for setting up the blockchain network and smart contracts for defining the fundamental functionalities. React Native, which offers an environment compatible with the Ethereum platform, is used to develop the EASEID's front end. NodeJS facilitates communication between the distributed application and the Ethereum framework. Variables, modifiers, states, and events—the fundamental building blocks of smart contracts—are built using the Solidity programming language. On the Testnet, the smart contracts are installed using a remix text network. The fundamental purpose of the Remix IDE is to create smart contracts that can be used both locally and worldwide. The MetaMask browser plugin, which adds a wallet as a browser extension, is used to connect to the Ethereum platform.

B. Performance Assessment

The Etherscan tool is used to assess the devised work's operational costs. It is an analytical tool that investigates the blockchain network's block. As a gas tracker for the Ethereum network, Etherscan keeps track of transactions, validates the effectiveness of smart contracts, and examines the status of processes. Gas as the cost is needed for the transaction's execution and is included in the cost blockchain network for Ethereum. The cost associated with carrying out a function in a blockchain network is measured in terms of Gas. The price of the Gas is determined by the miners based on supply and

Article Received: *14 August 2023* ***Revised:*** *04 October 2023* ***Accepted:*** *20 October 2023*

demand. The price is determined by the manner in which the execution, deployment, and transfer of the transaction on the blockchain network. Gas typically comprises two parameters: price and limit. The user's willingness to complete a transaction determines the limit, which is indicated as ‘gwei’. Smart contract and transaction execution require a significant amount of computational power over the blockchain community. The smart contracts are implemented on TestRPC in the proposed work, and details of all completed tasks are gathered by the Etherscan tool. The operational costs of the suggested method on the TestRPC-based Ethereum blockchain network and employing Remix platforms are displayed in Table 1. The entity that requests the execution of the smart contract function is referred to as the caller in Table 1. In this case, four operations, namely considered are register( ),generate( ),issue( ), and verify( ). Using the distributed application, the proposed blockchain network implements these features and computes the price of each operation's Gas.

TABLE I. OBTAINED OPERATIONAL COST (GAS COST)

Caller	Module	Remix	TestRPC
Healthcare/Users	*register*( )	0.000782	0.000396
Healthcare	*generate*( )	0.000805	0.000489
Healthcare	*issue*( )	0.000865	0.000602
Healthcare/Users	*verify*( )	0.000356	0.000287

The proposed system's operational costs are shown in Figure 6 when they are implemented on the Ropstern network's Ethereum blockchain-based system. Due to the processing necessary to complete the medical certificate generation process transactions, the suggested application raises the total Ether as the number of medical certificates increases. The transaction cost and execution cost of the suggested strategy are shown in Table 2. Any blockchain network often needs a reasonable amount of time to read information from a system application or an EASEID. The cost associated with executing a transaction is known as the transaction cost. Contrarily, execution cost is the full expense associated with adding the newly formed block having the blockchain structure containing numerous transactions. Table 2 illustrates the cost analysis of the proposed application's smart contract features in terms of transaction and execution costs.

A key drawback is the need to remember which clusters are "forbidden" when looking for a workaround. With normal routing, such a problem should not arise: it is obvious that for any pair of numbers $A_{src} = ab, A_{dest} = cd$, it is impossible when the component a or c refers to the "forbidden". As for components b and d, they are important only for intra-cluster routing.

TABLE II. COST ASSESSMENT FOR SMART CONTRACTS

Module	*Transaction cost*	*Computational cost*
request( )	2124420	1606341
generate( )	746168	426104
issue( )	1204620	988774
verify( )	965338	819928

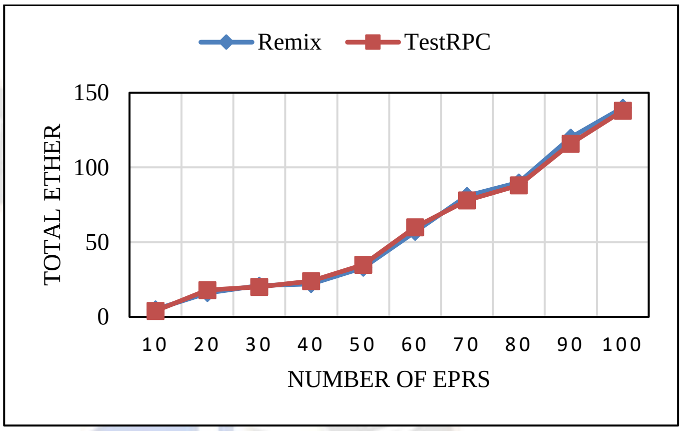

Figure 6. Total Ether consumption

Table 3 displays the performance of the suggested application when latency and processing time, two non-functional factors, are taken into account. The findings show the differences between the suggested plan with and without the deployment of a blockchain. Due to its internal operations, such as mining, crypto hash evaluation, transaction, block construction, and adding the new block to the blockchain network, the system installed in a blockchain platform consumes more time than a system without a blockchain.

TABLE III. PERFORMANCE ASSESSMENT

Fog-based Blockchain-enabled Platform	**Attributes (in Sec.)**	**Modules**	
		***Request_EPR*()**	***Verify_EPR*()**
Yes	Latency	4.12	3.20
Yes	Computational Time	2.76	3.05
No	Latency	10.86	8.12
No	Computational Time	9.23	7.42

The proposed work is also contrasted with previous research, including BinDaaS [37], data allocation using blockchain [38], and distributed healthcare networks [39]. Different facets of the proposed job, such as latency, throughput, and response time, are assessed and contrasted with similar efforts. The performance, robustness, and efficiency of the proposed work are assessed in comparison to those of the current works in this comparative analysis. The proposed work performance is compared to the existing works and evaluated based on the

Article Received: *14 August 2023* ***Revised:*** *04 October 2023* ***Accepted:*** *20 October 2023*

delay involved in creating and confirming medical certificates, as illustrated in Figure 7. The latency parameter calculates the amount of time that has passed between when a user starts a transaction and when it is added to the blockchain network as a block. The BinDaaS requires extra time to produce the forecast for the medical records. In this phase, the relevant entities generate the prediction result after first confirming that the message sought by the other entities is accurate. In order to move the request from the blockchain network to the cloud virtual machine, there is a network propagation delay involved in the data allocation with blockchain work. Furthermore, in the considered architectures, the principles of Fog computing are not taken into account. The inherent limitations of Cloud computing, i.e., ingress traffic and incurred latency are high due to the physical gap that exists between Cloud servers and IoMT. However, these limitations are subdued with the introduction of the Fog layer between the Cloud servers and IoMT. Nonetheless, this introduction of Fog computing literally brought the computations closer to the end devices and thus, reduced the computation time, response time and latency.

Due to the use of sophisticated authentication methods, the distributed hospital network model necessitates more delay than BinDaaS and data allocation using blockchain. As a result, processing medical papers required increased latency in the existing operations. In contrast to current works, the proposed work requires less latency time for processing medical records. The suggested effort speeds up the entire process of creating and verifying medical certificates by creating distinctive IDs for the medical documents. The proposed work has less latency than the existing literature, as shown in Figure 7. Therefore, compared to previous works, the proposed work offers a more reliable alternative for handling medical documents.

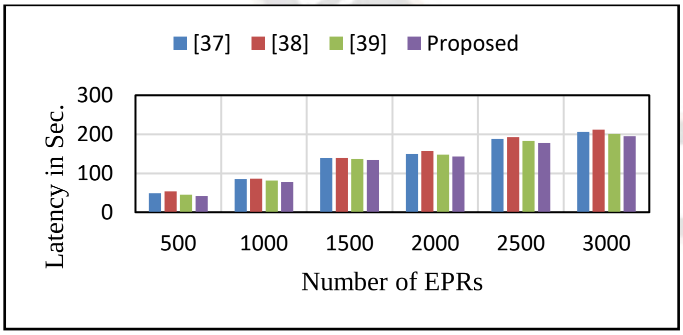

Figure 7. Performance analysis for latency

The suggested work performance is evaluated based on throughput and contrasted with the current works, as illustrated in Figure 8. The throughput counts how many transactions a user completes in a specific period of time. Compared to previous work, the suggested architecture has a higher throughput for processing transactions started at random by users. The throughput of the existing works is lower than the throughput of the proposed work because they need a two-step process to execute transactions for various users. Additionally, because the proposed design only permits the verified user to initiate the transaction, it directly executes the transaction in response to the user's request.

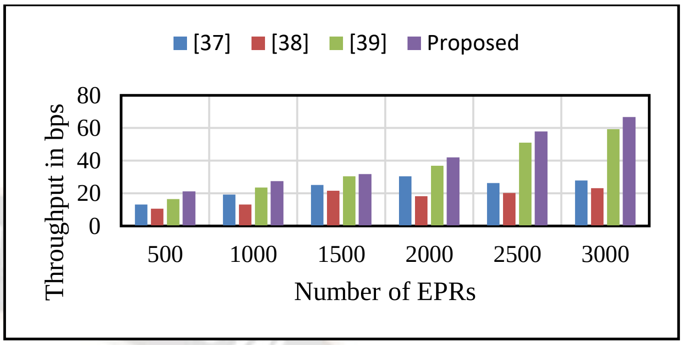

Figure 8. Performance analysis for Throughput

Thus, it operates more quickly. The performance of the proposed system is compared to the existing work by taking into account the response time necessary for processing the medical records, as shown in Figure 9. The response time takes into account the time needed for the system to complete the transaction and generate the response in accordance with the request. On the suggested system, the processing time for medical files of various sizes is measured. The proposed effort, however, takes less time to respond than the current work to process the medical data on the system. The current work entails additional overhead for handling medical records processing, which takes longer than the suggested system.

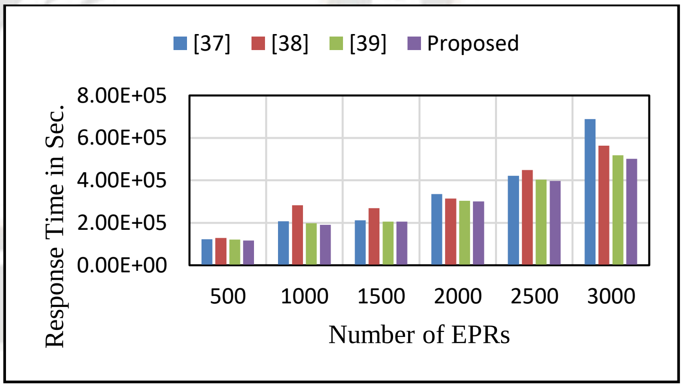

Figure 9. Performance analysis for Response Time

C. Security Analysis

In order to protect patient privacy in the healthcare sector, the suggested model makes use of ANN and blockchain to enable the FL approach to identify harmful users. The improvement in the suggested paradigm is demonstrated by the security study that follows.

1) The proposed paradigm includes the blockchain concept, which offers a tamper-resistant environment. The only operations that can be performed on the data in the healthcare system are added and search. In the proposed system, data deletion and change are not permitted.
2) FL, which incorporates numerous learning outcomes by the local model, can guarantee the accuracy of the given model.
3) The FL-block can withstand a wide range of assaults, even poisonous ones.
4) The proposed model's distributed environment prevents a single point of failure.

VI. CONCLUSIONS

In this paper, a cutting-edge method for privacy protection based on blockchain integration and deep learning was developed. In this study, three processed datasets were categorised using an ANN-based deep learning architecture. The proposed approach has been used to identify normal and atypical users. Besides, this study offers a revolutionary blockchain-based distribution architecture with privacy-preserving features for healthcare systems. A distributed application for creating and accessing medical certifications is offered by the suggested architecture. It uses a variety of smart contracts to register users, create certificates, validate users and certificates, stop attacks, and grant access to users. To assess the effectiveness of the suggested scheme, a number of experimental tests are carried out, and a number of metrics, including latency, processing time, throughput, and response time, are taken into account.

The future of a completely digital healthcare infrastructure will be shaped by the exchange of Electronic Patient Records (EPR). The urgent problems of data provider centrality and poor interoperability must, however, be addressed. These issues need to be resolved right away. EPR sharing on the blockchain can reduce these problems, yet current remedies cannot counteract malicious propagation. In this work, we emphasise the proposal of an effective EPRChain-based EPR sharing system and the spread of anti-malicious ideas. In particular, we suggest a universal designated verifier signature proof without pairing (UDVSP) plan and create a cordial and well-matched smart contract, then EPRChain's design specifications. Our proposals (such as EPRChain and UDVSP) are in accordance with the results of the final security study and performance assessment.

Making EPRChain feature-rich will include providing several plug-in designs, like authentication, access control, storage, and so forth), and expanding our UDVSP system to support numerous specified applications in the real world, such as verifiers.

REFERENCES

[1] Q. Wang, X. Zhu, Y. Ni, L. Gu, and H. Zhu, "Blockchain for the IoT and industrial IoT: A review," Internet Things, vol. 10, 2020, Art. no. 100081.

[2] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," IEEE Signal Process. Mag., vol. 37, no. 3, pp. 50–60, May 2020.

[3] B. Chen, L. Wu, N. Kumar, K.-K. R. Choo, and D. He, "Lightweight searchable public-key encryption with forward privacy over IIoT out-sourced data," IEEE Trans. Emerg. Topics Comput., vol. 9, no. 4, pp. 1753–1764, Oct.–Dec. 2021.

[4] A. Lakhan et al., "Smart-contract aware ethereum and client-fog-cloud healthcare system," Sensors, vol. 21, no. 12, 2021, Art. no. 4093.

[5] B. Chen, L. Wu, H. Wang, L. Zhou, and D. He, "A blockchain-based searchable public-key encryption with forward and backward privacy for cloud-assisted vehicular social networks," IEEE Trans. Veh. Technol., vol. 69, no. 6, pp. 5813–5825, Jun. 2020.

[6] Lakhan, M. A. Mohammed, S. Kadry, K. H. Abdulkareem, F. T. Al-Dhief, and C.-H. Hsu, "Federated learning enables intelligent reflecting surface in fog-cloud enabled cellular network," Peer J. Comput. Sci., vol. 7, 2021, Art. no. e758.

[7] X. Xu et al., "Privacy-preserving federated depression detection from multi-source mobile health data," IEEE Trans. Ind. Informat., vol. 18, no. 7, pp. 4788–4797, Jul. 2022.

[8] T. Hai et al., "DependData: Data collection dependability through three-layer decision-making in BSNs for healthcare monitoring," Inf. Fusion, vol. 62, pp. 32–46, 2020.

[9] 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 IEEE Global Communications Conference, GLOBECOM 2018, Abu Dhabi, United Arab Emirates, December 9-13, 2018, pp. 1–6, IEEE, 2018.

[10] C. Lin, D. He, X. Huang, K. R. Choo, and A. V. Vasilakos, "Bsein: A blockchain-based secure mutual authentication with fine-grained access control system for industry 4.0," J. Netw. Comput. Appl., vol. 116, pp. 42–52, 2018.

[11] C. Lin, D. He, N. Kumar, X. Huang, P. Vijayakumar, and K. R. Choo, "Homechain: A blockchain-based secure mutual authentication system for smart homes," IEEE Internet Things J., vol. 7, no. 2, pp. 818–829, 2020.

[12] C. Lin, D. He, X. Huang, M. K. Khan, and K. R. Choo, "DCAP: A secure and efficient decentralized conditional anonymous payment system based on blockchain," IEEE Trans. Inf. Forensics Secur., vol. 15, pp. 2440–2452, 2020.

[13] A. Dubovitskaya, Z. Xu, S. Ryu, M. Schumacher, and F. Wang, "Secure and trustable electronic medical records sharing using blockchain," in AMIA 2017, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 4-8, 2017, AMIA, 2017.

[14] M. Usman and U. Qamar, "Secure electronic medical records storageage and sharing using blockchain technology," Procedia Computer Science, vol. 174, pp. 321–327, 2020.

[15] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "Medrec: Using blockchain for medical data access and permission management," in 2nd International Conference on Open and Big Data, OBD 2016, Vienna, Austria, August 22-24, 2016 (I. Awan and M. Younas, eds.), pp. 25–30, IEEE Computer Society, 2016.

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

[17] J. Baek, R. Safavi-Naini, and W. Susilo, "Universal designated verifier signature proof (or how to efficiently prove knowledge of a signature)," in Advances in Cryptology - ASIACRYPT 2005, 11th International Conference on the Theory and Application of Cryptology and Information Security, Chennai, India, December 4-8, 2005, Proceedings (B. K. Roy, ed.), vol. 3788 of Lecture Notes in Computer Science, pp. 644–661, Springer, 2005.

[18] X. Chen, G. Chen, F. Zhang, B. Wei, and Y. Mu, "Identity-based universal designated verifier signature proof system," Int. J. Netw. Secur., vol. 8, no. 1, pp. 52–58, 2009.

[19] H.-Y. Lin, "Secure universal designated verifier signature and its variant for privacy protection," Information Technology and Control, vol. 42, no. 3, pp. 268–276, 2013.

[20] Sharma, Pratima, et al. "Blockchain-based privacy preservation for IoT-enabled healthcare system." *ACM Transactions on Sensor Networks* 19.3 (2023): 1-17.

[21] Lin, Chao, Xinyi Huang, and Debiao He. "Efficient Blockchain-Based Electronic Medical Record Sharing with Anti-Malicious Propagation." *IEEE Transactions on Services Computing (2023)*.

[22] Alzubi, Jafar A., et al. "Cloud-IIoT-based electronic health record privacy-preserving by CNN and blockchain-enabled federated learning." *IEEE Transactions on Industrial Informatics* 19.1 (2022): 1080-1087.

[23] Singh, Saurabh, et al. "A framework for privacy-preservation of IoT healthcare data using Federated Learning and blockchain technology." *Future Generation Computer Systems* 129 (2022): 380-388.

[24] Stephanie, Veronika, et al. "Trustworthy privacy-preserving hierarchical ensemble and federated learning in healthcare 4.0 with blockchain." *IEEE Transactions on Industrial Informatics* (2022).

[25] Passerat-Palmbach, Jonathan, et al. "Blockchain-orchestrated machine learning for privacy preserving federated learning in electronic health data." *2020 IEEE international conference on blockchain (Blockchain)*. IEEE, 2020.

[26] Gupta, Vagisha, Shelly Sachdeva, and Subhash Bhalla. "A novel deep similarity learning approach to electronic health records data." *Ieee Access* 8 (2020): 209278-209295.

[27] Zaman, Shakila, et al. "Thinking out of the blocks: Holochain for distributed security in iot healthcare." *IEEE Access* 10 (2022): 37064-37081.

[28] Ismail, Walaa N., et al. "CNN-based health model for regular health factors analysis in internet-of-medical things environment." *IEEE Access* 8 (2020): 52541-52549.

[29] Jia, Bin, et al. "Blockchain-enabled federated learning data protection aggregation scheme with differential privacy and homomorphic encryption in IIoT." *IEEE Transactions on Industrial Informatics* 18.6 (2021): 4049-4058.

[30] Li, Dun, et al. "Blockchain for federated learning toward secure distributed machine learning systems: a systemic survey." *Soft Computing* 26.9 (2022): 4423-4440.

[31] Qu, Youyang, et al. "Decentralized privacy using blockchain-enabled federated learning in fog computing." *IEEE Internet of Things Journal* 7.6 (2020): 5171-5183.

[32] Rahman, M. A., Hossain, M. S., Islam, M. S., Alrajeh, N. A., & Muhammad, G. "Secure and provenance enhanced internet of health things framework: A blockchain managed federated learning approach." *Ieee Access* 8 (2020): 205071-205087.

[33] Xu, J., Lin, J., Liang, W., & Li, K. C. "Privacy preserving personalized blockchain reliability prediction via federated learning in IoT environments." *Cluster Computing* 25.4 (2022): 2515-2526.

[34] Lakhan, A., Mohammed, M. A., Kozlov, S., & Rodrigues, J. J. "Mobile-fog-cloud assisted deep reinforcement learning and blockchain-enable IoMT system for healthcare workflows." *Transactions on Emerging Telecommunications Technologies* (2021): e4363.

[35] J. Baek, R. Safavi-Naini, and W. Susilo, "Universal designated verifier signature proof (or how to efficiently prove knowledge of a signature)," in Advances in Cryptology - ASIACRYPT 2005, 11th International Conference on the Theory and Application of Cryptology and Information Security, Chennai, India, December 4-8, 2005, Proceedings (B. K. Roy, ed.), vol. 3788 of Lecture Notes in Computer Science, pp. 644–661, Springer, 2005

[36] McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017, April). Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics* (pp. 1273-1282). PMLR.

[37] P. Bhattacharya, S. Tanwar, U. Bodke, S. Tyagi, and N. Kumar. 2020. BinDaaS: Blockchain-based deep-learning as-a-service in healthcare 4.0 applications. IEEE Transactions on Network Science and Engineering (2020).

[38] W. Yanez, R. Mahmud, R. Bahsoon, Y. Zhang, and R. Buyya. 2020. Data allocation mechanism for Internet of Things systems with blockchain. IEEE Internet of Things Journal 7, 4 (2020), 3509–3522.

[39] A. Yazdinejad, G. Srivastava, R. M. Parizi, A. Dehghantanha, K.-K. R. Choo, and M. Aledhari. 2020. Decentralized authentication of distributed patients in hospital networks using blockchain. IEEE Journal of Biomedical and Health Informatics 24, 8 (Aug. 2020), 2146–2156. DOI:10.1109/JBHI.2020.2969648.