I. J. Education and Management Engineering, **2025, 6, 37-52**
Published Online on December 8, 2025 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijeme.2025.06.04

Blockchain-Based Smart Medical Privacy-Preserving Contract System for Secure Electronic Health Records Management

B. Vasantha Rani*
School of CSE, Lovely Professional University, Phagwara, Punjab-India
Vignan's Institute Of Engineering For Women, Kappujagarajupeta
E-mail: badagalavasantha06@gmail.com
ORCID iD: https://orcid.org/0000-0001-7403-2636
*Corresponding Author

Parminder Singh
School of CSE, Lovely Professional University, Phagwara, Punjab-India
E-mail: Parminder.16479@lpu.co.in
ORCID iD: https://orcid.org/0000-0002-0750-6309

Received: 12 January, 2025; Revised: 15 April, 2025; Accepted: 22 June, 2025; Published: 08 December, 2025

Abstract: The rising frequency of data breaches and unauthorized access in healthcare systems has heightened concerns over privacy protection in Electronic Health Records (EHRs), especially when third parties, such as insurance companies, are involved. Traditional EHR solutions are often inadequate in safeguarding data confidentiality during multi-user access. This paper presents the Blockchain-based Smart Medical Privacy-preserving Contract System (BSMPCS), a novel framework designed to address these privacy challenges. By integrating Bitcoin smart contracts and homomorphic encryption, BSMPCS ensures secure management of EHRs while restricting access to only authorized entities. The decentralized and immutable characteristics of blockchain enhance data integrity, while smart contracts prevent unauthorized disclosure of sensitive health information to insurance companies. Additionally, homomorphic encryption enables insurance claim verifications without exposing the actual health records, preserving both patient identity and medical privacy. Unlike conventional systems, BSMPCS eliminates the need for third-party intermediaries, significantly reducing the risk of data leaks. Combining blockchain and advanced cryptographic methods, this framework provides a robust, privacy-preserving solution suitable for modern healthcare systems.

Index Terms: Electronic Health Records (EHR), Blockchain Smart Contracts, Homomorphic Encryption, Privacy Preservation, Cryptographic Computing

1. Introduction

At first, SMPC came about as a decentralized way to solve the known puzzle called the Millionaires' Problem, helping to compute a function without showing the inputs [1]. After that, the concept was further developed by Goodrich et al. to let many parties collaborate on computations, which provides a useful method for joint work in unsecure environments. Private data of several companies in SMPC is processed to calculate a function, yet only the outcome is shown and other information will not be revealed [2].

In the healthcare sector, Electronic Healthcare Information holds medical reports, prescriptions, pictures, as well as important details such as names, age, gender, and social security numbers. Since this information needs to be shared with insurance companies and research centers, it creates important social privacy difficulties. Centralized EHI storage and management systems do not always keep information safe and may easily suffer from unauthorized access. Because of this, cloud-based services have become viable, as they allow storage and access to grow as required [3]. Nevertheless, they remain vulnerable to a single weak area, data altering, and privacy leaks because they depend on third parties [4].

In addition, most access control systems, such as RBAC, ask for complicated rules and policies to be set up ahead of time based on different user roles. Since the model concentrates power, it is susceptible to privilege abuse and unapproved access of all the data [5]. Updating the system's access rules can leave it vulnerable to attacks since the main authority is always a one-point threat.

Because blockchain technology is decentralized, unchangeable, and safe, it can serve as a useful alternative to centralized management of EHI [7,8]. As a result of smart contracts being part of blockchain, people cannot gain access unless they are verified and authorized by different stakeholders, whose decisions are based on policies that cannot be changed by a single authority. ABAC mechanisms are also useful, since they allow creating access policies depending on user attributes, offering more flexibility.

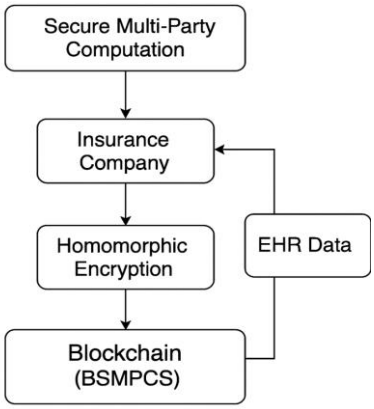

Fig. 1. BSMPCS Framework for Secure and Privacy-Preserving EHR Management

The BSMPCS framework is explained in this figure-1, which uses Secure Multi-Party Computation (SMPC), homomorphic encryption, and blockchain to secure and protect Electronic Health Records (EHRs). Because of homomorphic encryption, EHR data is protected and can be processed without letting anyone access it. Next, the encrypted data is kept on a blockchain network, which makes it secure and unchangeable. Smart contracts make it possible to interact with insurance companies while validating the information without disclosing patients' names or private health records. Thanks to this architecture, privacy in EHRs is improved by not having single major weaknesses and blocking unapproved access.

BSMPCS, a blockchain-based method, is our proposal to face the challenges of keeping information private and making sure EHR can be accessed only by authorized users. Homomorphic encryption method is used in this model to secure EHR data before it is stored on the blockchain. Computations on insurance verification are done on the encryption itself, so the private data stays hidden. Also, patients are able to keep their information and view its decrypted results when they want. This makes it possible to keep data private and at the same time give it value.

Section 2 gives an overview of the major technologies that are part of our research. The third section explains the BSMPCS framework, and Section 4 goes further to detail how the study was conducted. Details of the experimental setup and the obtained results can be found in Section 5, and conclusion and future scope are prepared in Section 6.

2. Literature Review

2.1 Blockchain-Based Secure Multiple Computation Scheme

Numerous secure multiple computation (MPC) techniques have been proposed in recent studies [5–8], with successful implementations in controlled environments [9]. These methods typically assume a semi-honest adversary model, where parties involved in MPC are assumed to follow the protocol but may attempt to glean additional information from the exchanged data. However, in reality, the adversaries may not always be so benign. Malicious actors, who intentionally disrupt the protocol to achieve desired outcomes, pose a significant challenge. Although there are several approaches to MPC that aim to secure computations under malicious adversary models [10–13], key challenges persist. For instance, when a participant gains a favorable outcome from the shared results, they may refuse to disclose them, thus preventing other participants from obtaining the benefits. The presence of malicious adversaries remains a continual threat, often hampering the protocol's proper execution. Thus, in practice, MPC systems must ensure that they are not only private but also resilient (protecting against denial-of-service attacks), fair (ensuring all parties receive the output or none), and efficient [14].

2.2 Smart Contract

A smart contract is a protocol designed to automatically execute, verify, or enforce the terms of an agreement in a secure and transparent manner. These contracts facilitate private, trustworthy transactions between unacquainted parties without the need for centralized authority or external enforcement mechanisms [15–17]. They provide self-executing and immutable algorithms that govern transactions, ensuring the conditions of the agreement are met before any action is taken.

2.3 Bitcoin

Bitcoin's decentralized platform allows users to interact with the blockchain and perform smart contract operations. The blockchain remains publicly accessible [8], with authenticated accounts enabling users to carry out blockchain transactions. Every operation and transaction is recorded in terms of gas units. The transaction cost is calculated by adding the execution cost (EC) and the transaction constant (TC), which represents the price of one gas unit:

$$PoT = (EC + TC) * P \tag{1}$$

Where PoT is the price of a transaction, EC is the execution cost, TC is the transaction constant, and P is the price of a single gas unit.

2.4 IPFS

The InterPlanetary File System (IPFS) is a protocol and network designed to provide a decentralized method for storing and exchanging hypermedia across a distributed file system [18]. It generates content-addressed identifiers for stored files, such as documents, images, and videos. These identifiers are formed using a Merkle Directed Acyclic Graph (Merkle DAG), with Public Key Infrastructure (PKI) used for identity management [19]. Each IPFS node is identified by a private key, and the node's identity is represented as a hash value:

$$KeyGen \rightarrow PUKey, PRKey \tag{2}$$

$$\text{NodeId} = \text{MultiHash (PUKey)} \tag{3}$$

The hashing scheme in IPFS is designed to ensure that hashes are deterministic, uncorrelated, unique, and one-way. The content identifier (CID) format in IPFS is structured as follows:

<cidv1>::=<multibase−prefix><cid−version><multicode−content−type><multihash−content−address>

Where the multiphase prefix encodes CIDs into different blocks, the CID version ensures upgradability, and the multicode content type identifies the data's format.

Blockchain technology, particularly in healthcare systems, leverages its decentralized, immutable nature to improve data privacy and security. Authors have employed blockchain to create systems for remote patient monitoring and emergency alerts [20]. These systems ensure data privacy by utilizing smart contracts to process patient data without exposing sensitive information, while homomorphic encryption enables secure insurance claim checks. This innovation significantly enhances real-time monitoring for healthcare providers and ensures that patient information remains confidential.

2.5 ApplicationinHealthcare

The concept of privacy-enhanced electronic health record (EHR) systems using blockchain and cryptographic technologies, such as smart contracts and homomorphic encryption, has gained significant traction. One notable example is the Blockchain-based Smart Multiple Party Computation Scheme (BSMPCS), which tackles the privacy concerns related to EHRs. With increasing incidents of data breaches and unauthorized third-party access, traditional EHR systems often struggle to ensure confidentiality when accessed by multiple parties. BSMPCS addresses these issues by utilizing blockchain’s decentralized nature and incorporating homomorphic encryption to perform operations on encrypted data. This ensures that sensitive patient information remains private, even when insurance companies or other entities need to verify certain claims or details.

Unlike conventional systems that involve third-party intermediaries, BSMPCS eliminates the risk of data exposure by empowering only authorized entities with access to the data. Furthermore, the use of smart contracts ensures that no patient data is disclosed to external parties unless specific conditions are met, thus offering robust protection for patient privacy. The application of this blockchain and cryptographic computing combination in EHR management represents a promising solution to the challenges of data security in healthcare systems [31].

Thus, the integration of blockchain and cryptographic technologies offers a promising framework for securing healthcare data, providing privacy, and improving operational efficiency while mitigating the risks of data exposure in electronic health records.

3. Proposed System

The proposed system, BSMPCS (Blockchain-based Smart Contract and Privacy-preserving Computation System), establishes a secure, decentralized, and scalable platform for the management and exchange of Electronic Health Records (EHRs) among authorized stakeholders in healthcare environments. It integrates blockchain, smart contracts, multi-agent systems, homomorphic encryption, and IPFS-based storage to ensure privacy, integrity, and fine-grained

access control. Both Attribute-Based Access Control (ABAC) and Role-Based Access Control (RBAC) are employed to enforce permissions, while each entity—patients, physicians, hospitals, insurers, and researchers—is authenticated and assigned predefined roles to prevent unauthorized disclosure of sensitive data.

Smart contracts play a central role in automating access policies, logging transactions, and ensuring transparency. In BSMPCS, Bitcoin smart contracts are used for immutable recording of consent, healthcare transactions, and verification processes, thereby guaranteeing traceability. Beyond simple access control, the smart contract logic supports complex scenarios such as updating patient records, resolving conflicts in shared access, and coordinating multi-party claim validations. Recognizing the importance of reliability, the system design incorporates security auditing and vulnerability testing to strengthen the resilience of deployed contracts in handling critical healthcare data.

Privacy is further reinforced through the integration of homomorphic encryption, which allows computations to be performed on encrypted data without decryption. Patients encrypt their EHRs before uploading, enabling physicians to generate treatment plans, insurers to verify claims, and researchers to analyze datasets without accessing raw records. While this approach ensures strong confidentiality, the system acknowledges the computational overhead, latency, and key management challenges associated with homomorphic encryption, and thus support hybrid approaches where only sensitive data is encrypted while routine metadata may remain unencrypted for efficiency. To complement blockchain storage, encrypted records are stored in the Interplanetary File System (IPFS), while blockchain maintains immutable hashes and access logs, ensuring verifiability and tamper resistance while supporting large-scale storage demands.

The system is designed to address scalability by combining blockchain with distributed IPFS storage and multi-agent parallelism. It is capable of managing large volumes of healthcare data and handling increasing user participation, with evaluation benchmarks focusing on throughput, latency, and transaction costs under simulated workloads. At the same time, BSMPCS recognizes the inherent limitations of Bitcoin in terms of speed and transaction fees, and therefore explores the adoption of layer-2 solutions and Sidechain to enhance scalability. Cost awareness is emphasized by analyzing transaction overheads in the healthcare context, an important factor for real-world deployment.

Compliance with regulations such as HIPAA and GDPR is an integral design goal. Smart contracts are programmed to enforce patient consent rules and role-specific permissions while blockchain audit trails facilitate accountability and regulatory inspections. Healthcare-specific concerns, including dynamic consent revocation, partial data sharing, and anonymous participation in research, are explicitly supported to align with privacy regulations. Furthermore, the environmental impact of blockchain adoption is acknowledged, as Bitcoin's proof-of-work consensus is energy-intensive. To address sustainability, the system considers energy-efficient alternatives and optimizations, ensuring that the adoption of blockchain in healthcare remains environmentally responsible.

In terms of usability, BSMPCS incorporates a role-adaptive user interface to support different types of stakeholders. Patients are provided with simple dashboards to manage consent and monitor access logs, doctors are given clinical interfaces with seamless access to encrypted medical data, insurers are enabled to validate claims without exposure to sensitive details, and researchers can request anonymized datasets through dedicated modules. These design choices aim to improve usability and adoption across diverse user groups, with planned usability studies to further refine the interface and user experience.

As depicted in Figure 2, the EHR privacy model illustrates the interaction of multiple authenticated agents through blockchain and smart contracts, ensuring transparency, traceability, and privacy in all healthcare-related exchanges.

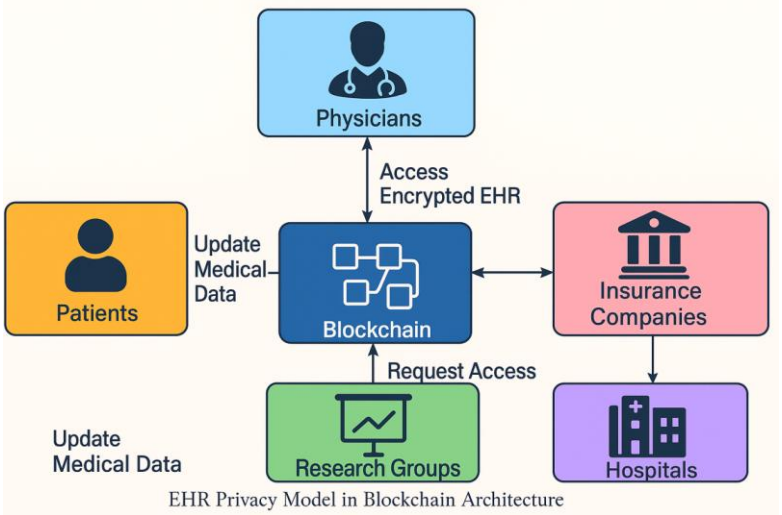

Fig. 2. EHR Privacy Model in Blockchain Architecture

Further illustrated in Figure 3 is the Homomorphic Encryption flow in the BSMPCS model. Patients collect their medical documents, which are then digitized and encrypted using homomorphic encryption. The encrypted EHRs are uploaded to the blockchain, making them accessible for processing without decryption. Physicians access encrypted

data, generate treatment plans, and submit prescription details. Insurance companies receive encrypted treatment data to verify claims without gaining access to plain-text EHRs. Patients can also link their encrypted EHRs with insurance providers for claim validation. Additionally, research groups can request access to encrypted datasets for analysis, ensuring patient anonymity. Hospitals oversee the entire system and can authorize or revoke access to entities to maintain privacy compliance.

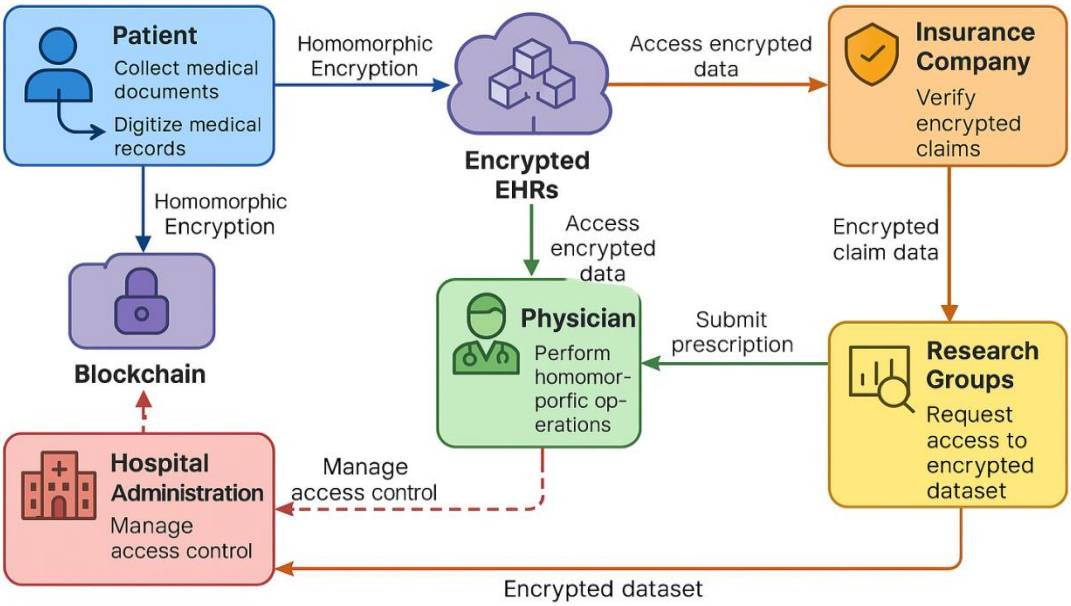

Fig. 3. Flow of Homomorphic Encryption Process in the BSMPCS Model

This figure-3 demonstrates the step-by-step flow of EHR handling—from patient-side encryption to data access and processing by doctors, insurance companies, and researchers. Homomorphic encryption allows secure operations on encrypted data, eliminating the need for decryption and enhancing privacy protection across the system.

4. Methodology

4.1 Homomorphic encryption

The chosen algorithm for homomorphic encryption is "ElGamal". In the case of ElGamal cryptosystem, a group G of order q with generator g and public key of (G, q, g, h) where h = gx and x is the secret key, then the encryption of a message m would be (gr, m, *hr*) . The homomorphic property refers to"

$$(m_1) \cdot (m_2) = (gr_1 + r_2, (m_1 \cdot m_2) hr_1, +r_2,) = (m_1 \cdot m_2) \quad (4)$$

When x is multiplied by g and the result is taken (mod p), you get Y.
And he will give his public key(Y, g, p) to Alice. After that, Alice prepares a message (M)to include in the packet.
A random value chosen is known as k. After that, she calculates the values of a and g.

- a is equal to the result of gk divided by p
- b equals ykM when modulo p is applied

then, each of them gets the cipher and decodes it using their respective keys.
When using Euler's theorem, the result is

$$M = x(\text{mod } p) \quad (5)$$

It happens this way because of
If the following is given:

$$a_1 = gk_1(\text{mod } p) \quad (6)$$

$$a_2 = gk_2\ (p) \tag{7}$$

Once you input for M, this formula is

$$b_1 = yk\ M\ (\text{mod } p) \tag{8}$$

b_2 is calculated by multiplying

$$y_k \text{ by } M_2\ (\text{mod } p) \tag{9}$$

After that, we obtain
The value of a is a1 multiplied by a_2, which is equal to the sum of gk and k_2

$$b = b_1 \times b_2 = ykM_1 \times yk\ M_2 \tag{10}$$

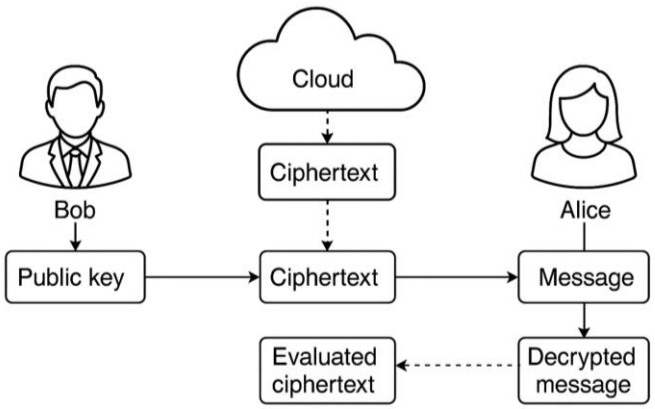

Fig. 4. Homomorphic Encryption process in HER-PP model

4.2 Psudeo Algorithm for Registering Patients

The smart contract will sign up patients, find out their data, and update their safety precautions in the blockchain. The pseudo algorithm is carried out in this way:

1. Begin
2. The details read have the patient's Aadhar number, name, address, phone number, blood group, insurance company, and emergency contact.
3. if the Aadhaar details in the array are found valid, then
4. if aadharnumber is found in the registered users then
5. Present("User already has an account")
6. Else
7. Include specific information in blockchain.
8. Else
9. Screen message ("Please make sure to provide valid user data")
10. End

It looks at the Aadhar number and sees if the details about the patient exist in the system. Should this be the case, the patient's details are then added to the blockchain.

4.3 Algorithm for Citizen's Need of Insurance and Medications

It is the contract's job to properly save the data of the patients who registered with their insurance. They look after the information on drugs that the insurance company pays for. If there were any necessary drugs that were not suggested in the doctor's list earlier, the insurance company allows authorized doctors to add the medications. It requires medication information is put into the function. It reviews if the selected drugs are included on your insurance coverage.

1. Begin
2. Get any company's information by sending its ID, name, and telephone number to the function.
3. If the Company is not registered then
4. Make the company's information part of the blockchain network.
5. Else
6. Make the visitor see an informative message with the text ("ID is already registered."). Give a different value for the ID)
7. End

Medications:

1. Begin
2. Treating the disease often calls for administration of prescribed medications.
3. address←fetching the address of the user
4. if the entity is an insurance company then
5. The Medic should use the details of the stored medications.
6. If (Medic. A step known as (Medication) takes place then.
7. The label should have a message that explains "Medications not covered by your insurance."
8. Else
9. Say("Your insurance covers the price for the medication")
10. Else
11. Remind them that ("Insurance companies are the only ones who can handle this task")
12. End

4.4 Pseudo Algorithm for claim for insurance

This function takes in input of patient's aadhar card number. It checks whether the entered aadhar number has already claimed for insurance or not.

1. Begin
2. Get the details (such as name or purpose) of medication.
3. Get the address of the User.
4. if (UADDRESS.insuranceCompany is true) then
5. Medic [] ← get data (on the medications)
6. If (Medic. There are (Medication) available.
7. The label "Medications that insurance does not cover" belongs on the display.
8. Else
9. Show an explanation of "covered procedures" on the website.
10. Else
11. According to the insurer, "The operation must be handled only by the insurance company"
12. End

4.5 Doctor's Pseudo Algorithm

All the doctor's activities planned in the schedule are handled by the contract. In the system, a doctor has to create his account in order to start any operation. Since the medical data is confidential, the patient must give permission before the doctor looks at it. Once the doctor has all the needed permissions, he comes up with a treatment id, treats the patient, and then puts the prescription and bill on the blockchain. He passes along the list of medications that are not included in the insurance company's details to them so they can add them.

1. Begin
2. Gathering the user's details → get the user details
3. The UADDRESS function is meant to retrieve the user's address.
If this user has been added to the system before, the address will be found.
5. When a person already registered, a message that reads "You are already registered" should appear.
6. Else
7. Service providers' information is managed best utilizing the blockchain.
8. End

4.6 Pseudo Algorithm for accessing the Patients data for treatment

This function reads the patient's Aadhar card number. It creates an OTP for the entered aadhar card number, using which doctor can access the patient's medical records.

1. Begin
2. The Paadhar tool ← fetches the Aadhar information about the patient
3. UADDRESS is found by using the function fetch on the User Address.
4. If the person is a doctor, then
5. OTP is calculated using Paadhar and keccak256.
6. Change your OTP to operate on blockchain networks.
7. Else
8. Only authorized doctors have the rights to access the patient's information.
9. End

4.7 Pseudo Algorithm for treatment and insurance

1. Begin
2. You should fetch the user's address from the UADDRESS variable.
3. If the role of the character is a doctor, then
4. The year Tid is given by (142317 * Paadhar)%1000003
5. Else
6. When clicking the icon ("Only the doctor can create a treatment id"), the treatment id appears.
7. End

4.8 Pseudo Algorithm for treating a patient

Its requires one to enter the patient's Aadhar card number. It confirms whether the person has claimed the insurance under the entered Aadhar number in the past.

1. Begin
2. User Address← call (get User Address)
3. if the status of being a doctor is set, then
4. It gets its information from a list that contains details about existing patients.
5. If all the patient's information is available, then
6. You should continue by making a new prescription and bill.
7. PatientDetails.add (Prescription)
8. Note on the chart ("Prescriptions and treatment plans have been updated for the patient")
9. Update blockchain
10. Else
11. One possibility would be to start with the registration process for the patient ("Display "Register the patient first")
12. Else
13. This is saying that ("Only a doctor can provide treatments to patients")
14. EndThe following modules create the MPS (multiple computation scheme) Contract, and this section explains what each module does.

Before starting, every user needs to register in this contract and pay at least the required deposit. It is only after completing this step that the participant can be chosen by users to give computing support in return for getting paid.

1) Reputation System: As part of this contract, every MPS node is given a reputation level. If a node in the MPS takes part and does its job well, its reputation value increases, or else it decreases. How useful MPS nodes become in the future is mostly decided by their reputational value.

2) The source of MPS node revenue is from the user fees and the penalties given to misbehaving nodes. The way rewards are given to nodes once each MPS task is finished shows the actual value of the nodes, according to the Incentive Mechanism.

3) Set the Quorum: To carry out the current calculation, users are allowed to select a certain number of MPS nodes depending on their reputation and deposits they have. If a company has a low reputation, it may still be chosen, but the chance of choosing that firm goes down very rapidly. The team should let the poor players chance to prove they can play better. It is important to consider someone just starting too. This reflects the truth in life.

4.9 Compute contract

Those who want to use multi-party computing sign up to Compute Contracts and add them to the blockchain. The Compute Contract provides details about the MPS architecture as well as which scripts are used per node. It happens round by round, and in each round, the contributors should supply precise information for the next stage of the process. The smart contract will ensure that the data supplied by the users is accurate. Also, the smart contract will notice if any member has missed sending the correct message in the first round. Afterwards, the blockchain will make sure the computation is allowed to progress. The length of the time limit can show when the block's peak will be reached. To begin, an actor will give an initial input to the MPS Contract so that MPS nodes start working on the calculation tasks.

At first, the users use the MPS nodes' public keys to encrypt their shares in the contract, and then pay the required fee. During the second and penultimate rounds, the MPS nodes add no worthful data to the blockchain just like the previous rounds. To finish, every MPS node handles and divides the production-related fees among themselves. From this perspective, the contract proves to be a dependable way for people to communicate. Put differently, this option serves as a real-time television broadcast.

4.10 MPS protocol

Let us look at some studies about current multi-party protocols that use contracts in the next group [40–42]. We are mainly concerned in our research with how robust and fair the model is. Checking the adherence of MPS protocols created using contracts is not included in our study. Malicious parties can also take part of the result of f' or calculate an amount that doesn't need to be there, as explained in [22]. As a result, each round's users and MPS will send their personal shares to the blockchain at the end of the round. As required, rewards and punishments are given by the Smart Contract. There are some duties the Users and MPS nodes should complete at the end of every cycle.

The purpose of BSMPSS uses blockchain and applies threshold contracts that work in rotating featured rounds. Time stamps on blockchain increase the possibility of different protocols running smoothly together. The Smart Contract will find and remove players who say anything unlawful during the up-next round. The system handles all these functions on its own and everyone can check them directly. This issue could not be managed easily because there was no single point to oversee the network. Due to special accounts and key pairs given to each member, chances of identity theft on the blockchain are very slim. There are some important aspects that demonstrate how powerful BSMPSS is, as you will see below.

a) Tracking anyone's reputation is possible because the ledger records all transactions made by each person. A reputation that has been built over a lengthy time is important for a person's chances of being selected. A deposit must be presented in the agreement when joining MPS as a strategy to protect against the Sybil Attack. Altogether, the attacker does not have the ability to use extra nodes to affect the protocol's operation. Should you stop the process at this stage, you may work on another subset and continue with it.

b) Users equally have to pay service fees by transferring a number of coins when they submit their tasks. The user's contribution will be checked and validated by the smart contract. Those who do not supply the right information within the set time will be immediately charged. The cost per service request ensures the security of the programme from users trying hacks.

c) The smart contracts also make use of the node's intermediate solution. When the node is selected, a valid message is needed, as if no response is provided, it will be moved to the corrupted set the next round. When the number of corrupt participants does not go beyond n-t, the protocol is put into effect.

The protocol will be set so that anyone violating the rules is not allowed to change tasks. The plan works like a (t, n)-theshold scheme, so that if t is well set, small errors will not disrupt the recovery of the secret. Participants may choose to belong to a new subset to put forward a different rule when the prior one is wrapped up. Therefore, it has been noticed that BSMPSS poses many challenges.

5. Experiment & Evaluation

Simply put, IPFS Storage, the insurance company, the pharmacist, the blockchain network, the patient, and the doctor are all parts of implementation. A prototype of Java has been built with the aid of Geth, which is the software for blockchain development. You can get both source codes on the Internet without paying. Because the blockchain lacks adequate storage, the CID is added to the blockchain, but every report is saved in IPFS version 4. IPFS Network is based on version 4 technology. There are Intel Core i5 7th Gen processors used in these computers: some have (i) 8 GB RAM, 500 GB local storage, and Windows x64 as the operating system, and some have (ii) 6 GB RAM, 500 GB local storage, and Ubuntu 16 as their operating system.

It is clear that the speed at which new blocks are made on the blockchain plays a big role in the performance of all blockchain applications. It usually needs about ten minutes for a Bitcoin transaction to happen. The organization has opted to use EOS blockchain. The EOS consensus method BFTDPoS allows the generation of a block every 0.5 seconds. MPC and Compute activities are included in the chain's BSMPCS work.

The PVSS data initiative process is the part of off-chain execution that takes the most time, since it means sending the Shamir secret and doing several asymmetric encryption calculations. The period it takes for MPC nodes to finish the Compute Contract Init phase mainly relies on how big the quorum is.

It can be observed that higher quorum size means the time taken for one round of BSMPCS in Compute Contract is expected to grow exponentially. It allows asymmetric computations to be repeated with additional nodes being added to a computing node's workload in each round of computations. In addition, since only a higher power polynomial can enable sharing secrets, the size of the created shares gets bigger and takes more time because of the asymmetric encryption. Deciding to have more MPC nodes for a given compute job would have a great impact on efficiency if the quorum exceeds 40.

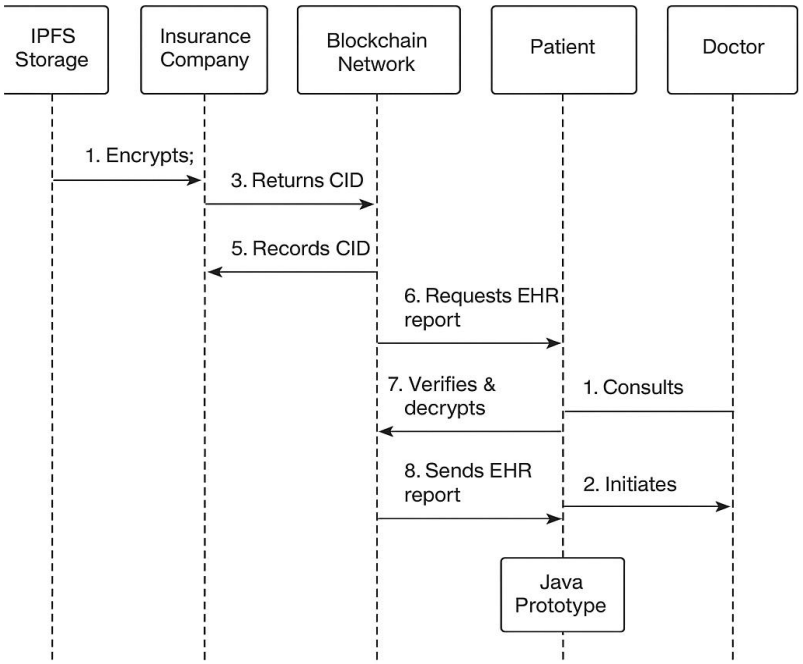

Fig. 5. Sequence diagram for EHR report in Healthcare System

This module collects the patient's records and hands them over to the data encoder to get the information in encoded form. These files are going to be updated on the blockchain and stored using IPFS technology. This information is provided on Fig. 6. It is apparent from the graphs that the process of getting new transactions costs more time than the process of receiving them.

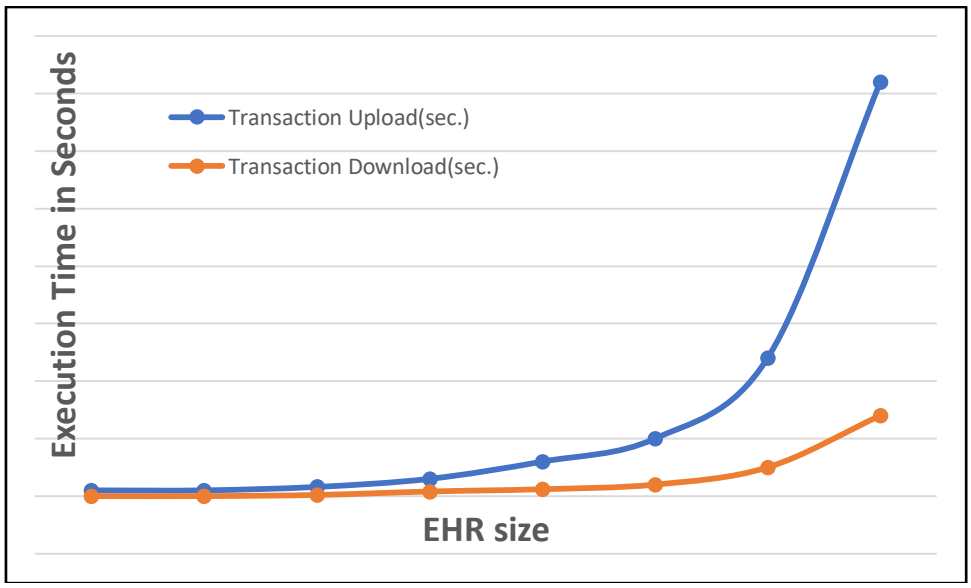

Fig. 6. Transaction Upload and Download Execution Time

a) As soon as a patient is registered, the request goes to a related doctor through the system. In order to fetch the data, he gains access by getting clearance from the blockchain's system. In addition, he produces a unique treatment ID for the procedure that is taking place now. As well as giving suggestions for drugs and tests, the doctor adds all the data to the blockchain.

b) In this module, both the hospital and the patients are covered by the insurance company along with the blockchain network. The patients need to register with the insurance company depending on the details of their medical emergency. The necessary details are checked by the hospital and the doctor with the insurer. When it is proven that the request is valid, the claim of the patient is denied; but if not, the information is processed and monetary benefits are given to the patient.

c) Patients should use the online pharmacy and give their doctor's prescription to purchase the required medicine online. The blockchain will incorporate the same details as well.Finally, the reports from the surveys are saved in IPFS by the people in the network. When reports are uploaded, every file gets a Content Identifier assigned to it. It is used as a reference to access files on the network. This created Content Identifier is saved directly in the blockchain.

d) The blockchain network module makes sure that the deals offered by several users are verified and then uses consensus methods to put together a fresh block. In the first step of mining, several blocks will be created with details of patients, doctors, treatment IDs, and insurance companies. It is visible that mining needs more time than the process of building a block. You can notice this in fig. 7.

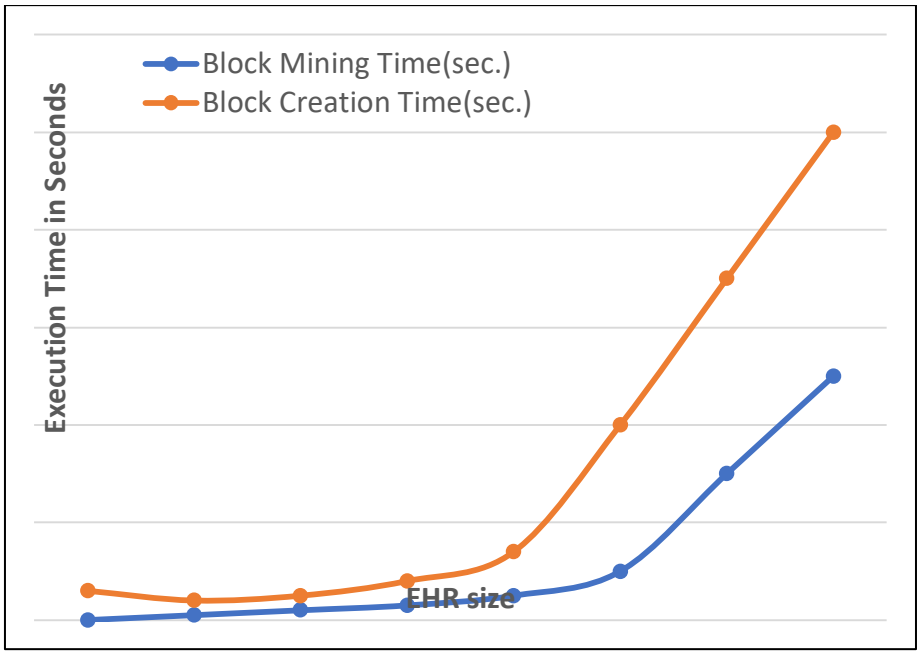

Fig. 7. Execution Time for Block Mining and Block Creation

Fig. 8 shows the time it takes peers to access (availability) a transaction for a range of report sizes. It seems sense that the access time would rise as the report's size rose. Furthermore, a transaction takes longer to complete the more peers participate.

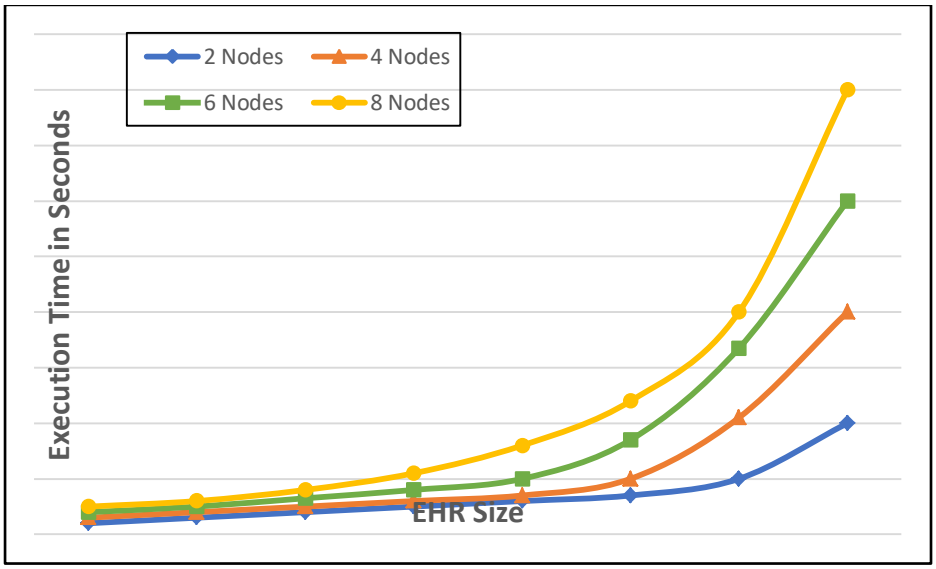

Fig. 8. Transaction Access Running Time vs. Number of Nodes

This section looks at how the new approach matches up with seven common methods in Health Record Information Systems focused on privacy. The comparison looks at frameworks set out by Azaria et al. [26], Omar et al. [24], Li et al. [25], Fan et al. [27], Zhang et al. [28], Shen et al. [29], and Uddin et al. [30].

It is evaluated using seven main security and privacy factors like tamper-proofing, protection of patient privacy, safeguarding confidential information, keeping data authorized, preventing denying used services, revoking access when needed, and searching the database. The summary of these metrics can be found in Table 1.

From the analysis, it is clear that the proposal is better than other schemes in all important aspects. When you use it, your EHR data remains secure because it deploys strong security mechanisms and privacy tools. Our solution is especially helpful in situations where other strategies do not work so well, especially for privacy, storing confidential data, and blocking access to information.

Table 1. Metrics

S.No	Metrics	Azaria [26]	Omar [24]	Li [25]	Fan [27]	Zhang [28]	Shen [29]	Uddin [30]	Proposed
1	Tamper – Proof	√	√	√	√	√	√	√	√
2	Privacy Preserving of Patient	✗	√	✗	√	✗	✗	✗	√
3	Data Privacy(Confidentiality)	✗	√	✗	✗	✗	✗	✗	√
4	Access Control	√	✗	✗	✗	√	√	√	√
5	Non – Repudiation	√	√	√	√	√	√	√	√
6	Access Revocation	✗	✗	✗	✗	√	✗	√	√
7	Block Search	√	✗	✗	√	√	√	√	√

5.1 Evaluations

All the necessary patient data is kept safe and private by the application. There are modules ready for each group, from the physicians and patients to the research workers, hospitals, and those handling insurance. The main aim of the EHR system is to make sure data collected and stored is private. To summarize, the model in Blockchain aims to block data tampering, prevent unlawful changes, and offer support for online healthcare systems. Homomorphic encryption and blockchain are used in this system to assure that data remains secure and private. The by-design rules in smart contracts make it possible for given entities to exchange data.

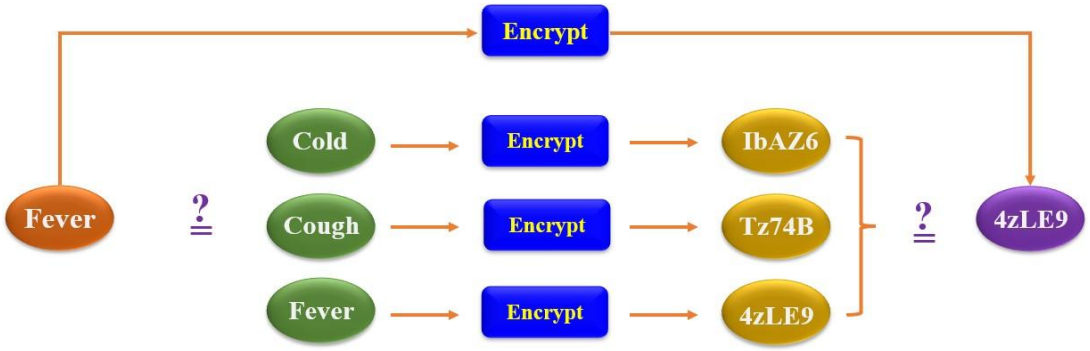

Fig. 9. HME working mechanism

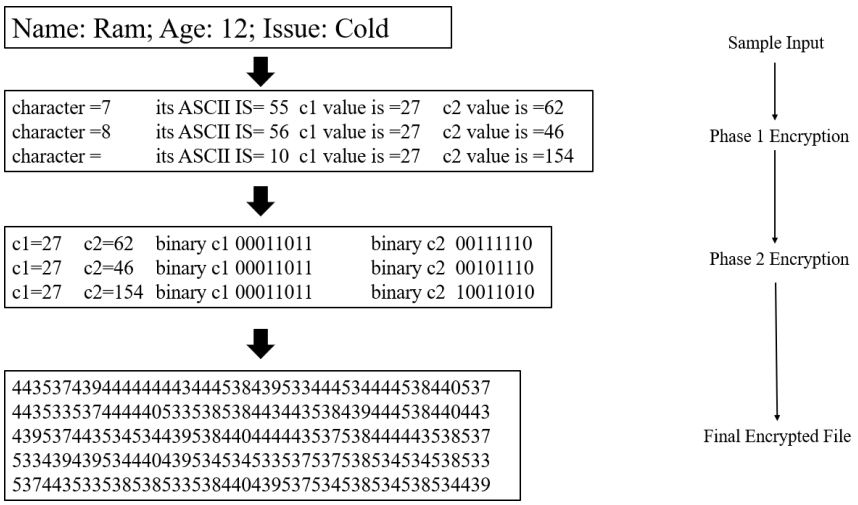

Fig. 10. Encryption process in ElGamal Encryption Algorithm

A test version of the system was established to find out how well it functions and how well the system handles privacy and security of the information included. The prototype was built by using Ethereum as the blockchain platform, go-Ethereum 7.5.4 for npm, and node 15.8.0 as its version. With version 5.1.66 of the Solidity Framework and IDE, you can deploy smart contracts, while writing the contracts in version 0.8.1 of the Solidity scripting language. When it comes to web3js for node queries on Ethereum [2], users should install version v1.2.0. Each of the six devices used to build the blockchain node features an Intel CPU of 2.30GHz and 8 GB of main memory. Different nodes were created in different systems including Windows 10 and Ubuntu 16. Encryption and decryption modules designed in the HME process of the EHR-PP system are created using Java 13.2, and Notepad++ is used for editing the code.

After getting the word “Fever”, the algorithm transforms it to “4zLE9” and then compares it with the encrypted information in the database.

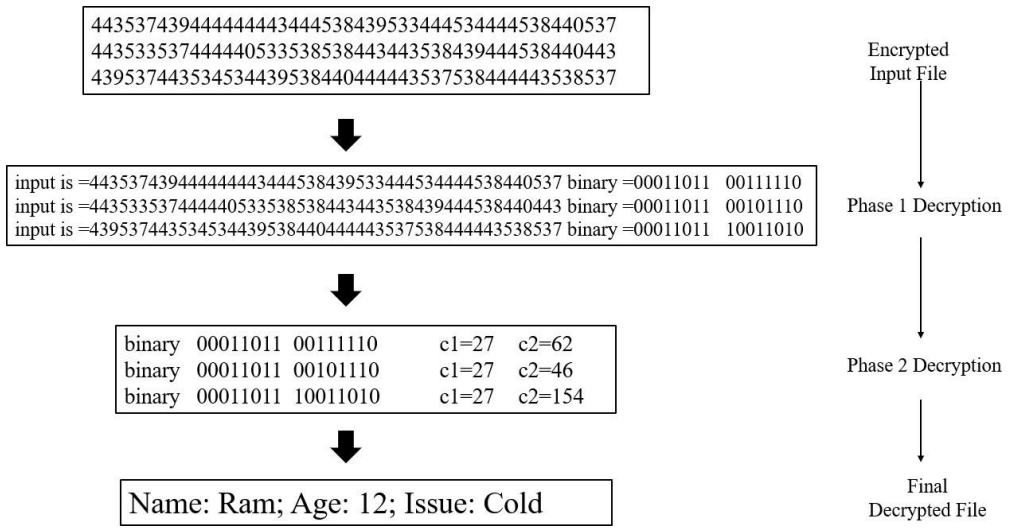

Fig. 11. Decryption process in ElGamal Decryption Algorithm

Table 2. Transaction and execution cost

S. No.	Operation Performed	Transaction cost		Executionc ost		Total cost(ine ther)
		In gas units	In ethers	In gas units	Inethers	
1.	Contract Deployment	1955973	0.2621004	1444301	0.1935363	0.4556367
2.	Addchemist	166589	0.0223229	142821	0.019138	0.0414609
3.	Adddoctor	207754	0.027839	182834	0.0244998	0.0523388
4.	Addinsurance Company	126755	0.0169852	104651	0.0140232	0.0310084
5.	Addpatient	208706	0.0279666	184170	0.0246788	0.0526454
6.	Getpatient Information	29325	0.0039296	7797	0.0010448	0.0049744
7.	UpdatePrecautions	44111	0.0059109	21367	0.0028632	0.0087741
8.	Getchemist	25707	0.0034447	4243	0.0005686	0.0040133
9.	Getinsurance Company	26489	0.0035495	5089	0.0006819	0.0042314
10.	AddMedications Notcoveredininsurance	65770	0.0088132	44114	0.0059113	0.0147245
11.	CreateTreatment-Id	21857	0.0029288	329	0.0000441	0.0029729
12.	TreatPatient	189040	0.0253314	165144	0.0221293	0.0474607
13.	Gettreatment Details	27482	0.0036826	5890	0.0007893	0.0044719
14.	Medication Includedininsurance	53756	0.0072033	31780	0.0042585	0.0114618
15.	Getdoctordetails	24375	0.0032663	2911	0.0003901	0.0036564

When the proposed framework is used, a smart contract is deployed and the amount of gas used defines its complexity. Table 1 shows the costs for every single operation in the setup, using Gwei as the standard. The total cost is reached by multiplying how many gas units were used by the price of each unit.

We need to do certain actions from the list on the chart to make the framework fully work. The total gas needed for all the operations comes to 0.7274322 ETH.

A comparison is made in this section between the proposed framework and existing ways to preserve privacy in HRI systems. During the assessment, main factors are checked, such as access control, revoking access, ensuring patient privacy, making the system accessible to patients, linking with analytics, and supporting smart choices.

The data in table 3 show the results of the comparison. Unlike the approaches being applied now, the suggested model integrates all the important features required to manage security and data access. It provides safe ways for users to access their health data and remove the access in the future, which boosts patient empowerment and uses machine learning to help healthcare workers make better choices.

Table 3. Performance Comparison of Privacy-Preserving Health Record Information Systems

Reference	Access Control	Access Revocation	Patient Privacy	Patient Data Access	Predictive Analytics	Intelligent Decision Support
Al-Sumaidaee et al. (2023)	✔	✔	✖	✖	✖	✔
Egala et al. (2023)	✔	✖	✖	✔	✖	✖
Abou El Houda et al. (2022)	✔	✖	✔	✔	✖	✖
Proposed Framework	✔	✔	✔	✔	✔	✔

6. Conclusion

It developed a framework that uses blockchain and a strong MPC scheme to deal with fairness and robustness in MPCs. A public reputation system is included, meaning that those with high scores are chosen first, so credible people play the biggest roles. We pointed out that protecting healthcare data is vital, mainly for privacy and security, and introduced an innovative system that encrypts the data in order to safeguard it at all times. Thanks to this encryption model, users are able to work on their encrypted data without first decrypting it. Connecting this model to blockchain, we created a more secure system that is better than the encryption and storage systems used commonly in the past. In order to decrease the possibility of altered results, the framework relies on a feature called smart contract that monitors each computation step. The setup encourages people to be honest since they can earn more as a result, while those who misbehave are disadvantaged. Due to the privacy issues of blockchain's public ledger, MPC was put in place to fix them. Going ahead, more research will concentrate on finding additional privacy benefits from MPC in healthcare-related blockchain projects, enhancing scalability, bringing in Machine Learning approaches, advancing Fully Homomorphic Encryption, and addressing issues related to healthcare regulation

References

[1] R. Cleve, ''Limits on the security of coin flips when half the processors are faulty,'' in Proc. 18th Annu. ACM Symp. Theory Comput. (STOC), 1986, pp. 364–369.
[2] J. Halpern and V. Teague, ''Rational secret sharing and multiparty computation: Extended abstract,'' in Proc. 36th Annu. ACM Symp. Theory Comput. (STOC), 2004, pp. 623–632.
[3] A. Lysyanskaya and N. Triandopoulos, ''Rationality and adversarial behavior in multi- party computation,'' in Advances in Cryptology— CRYPTO. Berlin, Germany: Springer, 2006, pp. 180–197.
[4] A. Groce and J. Katz, ''Fair computation with rational players,'' in Advances in Cryptology—EUROCRYPT. Berlin, Germany: Springer, 2012, pp. 81–98.
[5] M. Nojoumian and D. R. Stinson, ''Socio-rational secret sharing as a new direction in rational cryptography,'' in Decision and Game Theory for Security. Berlin, Germany: Springer 2012, pp. 18–37.
[6] G. Asharov, R. Canetti, and C. Hazay, ''Towards a Game Theoretic View of Secure Computation,'' in Advances in Cryptology—EUROCRYPT. Berlin, Germany: Springer 2011, pp. 426–445.
[7] S.Nakamoto.(2008).Bitcoin:A Peer-to-Peer Electronic Cash System. [Online]. Available: https://bitcoin.org/bitcoin.pdf
[8] M. Andrychowicz, S. Dziembowski, D. Malinowski, and Ł. Mazurek, ''Fair two-party computations via Bitcoin deposits,'' in Financial Cryptography and Data Security. Berlin, Germany: Springer, 2014, pp. 105–121.
[9] M. Andrychowicz, S. Dziembowski, D. Malinowski, and L. Mazurek, ''Secure multiparty computations on Bitcoin,'' in Proc. IEEE Symp. Secur. Privacy, May 2014, pp. 443–458.
[10] I. Bentov and R. Kumaresan, ''How to Use Bitcoin to design fair protocols,'' in Advances in Cryptology—CRYPTO. Berlin, Germany: Springer, 2014, pp. 421–439, 2014.
[11] R. Kumaresan and I. Bentov, ''How to Use Bitcoin to incentivize correct computations,'' in Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), 2014, pp. 30–41.
[12] R. Kumaresan, T. Moran, and I. Bentov, ''How to use Bitcoin to play decentralized poker,'' in Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), 2015, pp. 195–206.
[13] R. Kumaresan, V. Vaikuntanathan, and P. N. Vasudevan, ''Improvements to secure computation with penalties,'' in Proc. ACM SIGSAC Conf. Comput. Commun. Secur. (CCS), 2016, pp. 406–417.
[14] A. Kiayias, H.-S. Zhou, and V. Zikas, ''Fair and robust multi-party computation using a global transaction ledger,'' in Advances in Cryptology— EUROCRYPT. Berlin, Germany: Springer, 2016, pp. 705–734.
[15] Juan Perez. Facebook, google launch data portability programs to all, 2008.
[16] Latanya Sweeney. k-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 10(05):557–570, 2002.

[17] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthu ramakrishnan Venkita subramaniam. l-diversity: Privacy beyond k-anonymity. ACM Transactions on Knowledge Discovery from Data (TKDD), 1(1):3, 2007.
[18] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and l-diversity. In ICDE, volume 7, pages 106–115, 2007.
[19] Arvind Narayanan and Vitaly Shmatikov. How to break anonymity of the netflix prize dataset. arXiv preprint cs/0610105, 2006.
[20] Yves-Alexandre de Montjoye, Cesar A Hidalgo, Michel Verleysen, and ´Vincent D Blondel. Unique in the crowd: The privacy bounds of human mobility. Scientific reports, 3, 2013.
[21] Cynthia Dwork. Differential privacy. In Automata, languages and programming, pages 1–12. Springer, 2006.
[22] Craig Gentry. Fully homomorphic encryption using ideal lattices. In STOC, volume 9, pages 169–178, 2009.
[23] Soujanya, D. and Ramana, K.V., 2021. Secured Surveillance Storage Model Using Blockchain. In Evolving Technologies for Computing, Communication and Smart World (pp. 249-263). Springer, Singapore
[24] Al Omar, A.; Rahman, M.S.; Basu, A.; Kiyomoto, S. MediBchain: A blockchain based privacy preserving platform for healthcare data, Security, Privacy, and Anonymity in Computation, Communication, and Storage, Guangzhou, China, 12–15 December, 2017; Wang, G., Atiquzzaman, M., Yan, Z., Choo, K.K., Eds.; Springer: Cham, Switzerland, 2017.
[25] Li, H.; Zhu, L.; Shen, M.; Gao, F.; Tao, X.; Liu, S. Blockchain-Based Data Preservation System for Medical Data. J. Med. Syst. 2018, 42, 1–13.
[26] Azaria, A.; Ekblaw, A.; Vieira, T.; Lippman, A. MedRec: Using blockchain for medical data access and permission management. In Proceedings of the 2016 2nd International Conference on Open and Big Data (OBD), Vienna, Austria, 22–24 August 2016; pp. 25–30.
[27] Fan, K.; Wang, S.; Ren, Y.; Li, H.; Yang, Y. MedBlock: Efficient and Secure Medical Data Sharing Via Blockchain. J. Med. Syst. 2018, 42, 1–11.
[28] Zhang, A.; Lin, X. Towards Secure and Privacy-Preserving Data Sharing in e-Health Systems via Consortium Blockchain. J. Med. Syst. 2018, 42, 1–18.
[29] Bingqing Shen , Jingzhi Guo and Yilong Yang, MedChain: Efficient Healthcare Data Sharing via Blockchain, Journal of Applied Science (MDPI), Volume 9(6), 1207, https://doi.org/10.3390/app9061207, 2019.
[30] Uddin, M.; Memon, M.S.; Memon, I.; Ali, I.; Memon, J.; Abdelhaq, M.; Alsaqour, R. Hyperledger Fabric Blockchain: Secure and Efficient Solution for Electronic Health Records. CMC Comput. Mater. Continua. 2021, 68, 2377–2397
[31] Peng, Z.; Xu, C.; Wang, H.; Huang, J.; Xu, J.; Chu, X. P2b-trace: Privacy-Preserving Blockchain-Based Contact Tracing to Combat Pandemics. In Proceedings of the 2021 International Conference on Management of Data, Xi'an, China, 20–25 June 2021; pp. 2389–2393.
[32] Alrebdi, N.; Alabdulatif, A.; Iwendi, C.; Lian, Z. SVBE: Searchable and verifiable blockchain-based electronic health records system. Sci. Rep. 2022, 12, 266. [CrossRef] [PubMed]
[33] Mondal, S.; Shafi, M.; Gupta, S.; Gupta, S.K. Blockchain Based Secure Architecture for Electronic Healthcare Record Management. GMSARN Int. J. 2022, 16, 413–426.
[34] Alrebdi, N.; Alabdulatif, A.; Iwendi, C.; Lian, Z. SVBE: Searchable and verifiable blockchain-based electronic health records system. Sci. Rep. 2022, 12, 266. [CrossRef] [PubMed]
[35] Mondal, S.; Shafi, M.; Gupta, S.; Gupta, S.K. Blockchain Based Secure Architecture for Electronic Healthcare Record Management. GMSARN Int. J. 2022, 16, 413–426.
[36] Cerchione, R.; Centobelli, P.; Riccio, E.; Abbate, S.; Oropallo, E. Blockchain's coming to hospital to digitalize healthcare services: Designing a distributed electronic health record ecosystem. Technovation 2023, 120, 798–805. [CrossRef]
[37] Chatterjee, A.; Pahari, N.; Prinz, A. HL7 FHIR with SNOMED-CT to Achieve Semantic and Structural Interoperability in Personal Health Data: A Proof-of-Concept Study. Sensors 2022, 22, 3756. [CrossRef] [PubMed]
[38] Jayabalan, J.; Jeyanthi, N. Scalable blockchain model using off-chain IPFS storage for healthcare data security and privacy. J. Parallel Distrib. Comput. 2022, 164, 152–167. [CrossRef]
[39] Ruan, P.; Dinh, T.T.A.; Lin, Q.; Zhang, M.; Chen, G.; Ooi, B.C. Revealing Every Story of Data in Blockchain Systems. ACM SIGMOD Rec. 2020, 49, 70–77. [CrossRef]
[40] Tith, D.; Lee, J.S.; Suzuki, H.; Wijesundara, W.M.A.B.; Taira, N.; Obi, T.; Ohyama, N. Application of blockchain to maintaining patient records in electronic health record for enhanced privacy, scalability, and availability. Healthc. Inform. Res. 2020, 26, 3–12.
[41] Mani, V.; Manickam, P.; Alotaibi, Y.; Alghamdi, S.; Khalaf, O.I. Hyperledger Healthchain: Patient-Centric IPFS-Based Storage of Health Records. Electronics 2021, 10, 3003.
[42] Chenthara, S.; Ahmed, K.; Wang, H.; Whittaker, F.; Chen, Z. Healthchain: A novel framework on privacy preservation of electronic health records using blockchain technology. PLoS ONE 2020, 15, e0243043.
[43] Sun, J.; Yao, X.; Wang, S.; Wu, Y. Blockchain-based secure storage and access scheme for electronic health records in IPFS. IEEE Access 2020, 8, 59389–59401.
[44] Verdonck, M.; Poels, G. Decentralized Data Access with IPFS and Smart Contract Permission Management for Electronic Health Records. In International Conference on Business Process Management; Springer: Cham, Switzerland, 2020; pp. 5–16.
[45] Ashizawa, N.; Yanai, N.; Cruz, J.P.; Okamura, S. Eth2Vec: Learning contract-wide code representations for vulnerability detection on ethereum smart contracts. Blockchain Res. Appl. 2022, 1, 100101.
[46] Chelsey C. Y. Hang, M. Batumalay, T D Subash, R. Thinakaran and B. Chitra, "Blockchain-based and IoT-based Health Monitoring App: Lowering Risks and Improving Security and Privacy" International Journal of Advanced Computer Science and Applications(IJACSA), 15(6), 2024. http://dx.doi.org/10.14569/IJACSA.2024.01506103
[47] Daraghmi, E.-Y.; Daraghmi, Y.-A.; Yuan, S.-M. MedChain: A Design of Blockchain-Based System for Medical Records Access and Permissions Management. IEEE Access 2019, 7, 164595–164613.

Authors Profiles

B. Vasantha Rani received her bachelor's degree from DIET, JNTUK in 2013. She attained her M.Tech degree from DIET, JNTUK in 2018. She is pursuing her Ph.D. in school of Computer Science and Engineering Department at Lovely Professional Unuversity from 2019. Presently, she is working as an Assistant Professor in the Department of IT in Vignan Institute of Engineering for Women (Autonomous). Her areas of interest include the Networking, Machine Learning, Deep Learning, and Blockchain Technology.

Parminder Singh (Member, IEEE) received the M. Tech. degree in computer engineering from Punjab Technical University, India, and the Ph.D. degree from Lovely Professional University, India, in 2019. He is currently working as an Associate Professor with the School of Computer Science and Engineering, Lovely Professional University. He has more than 30 articles in SCI/SCIE, Scopus indexed journals, conferences, and book chapters. His research interests include machine learning, deep learning, blockchain, cloud/fog/edge computing, network security, and Web services. He is an Active Member of IEEE. He has been the Session Chair and an Advisory Member for various international conferences.

How to cite this paper: B. Vasantha Rani, Parminder Singh, "Blockchain-Based Smart Medical Privacy-Preserving Contract System for Secure Electronic Health Records Management", International Journal of Education and Management Engineering (IJEME), Vol.15, No.6, pp. 37-52, 2025. DOI:10.5815/ijeme.2025.06.04