





Privacy-Preserving Blockchain-Based EHR Using ZK-Snarks

R. Anusuya^(✉) , D. Karthika Renuka , S. Ghanasiyaa, K. Harshini, K. Mounika, and K. S. Naveena

PSG College of Technology, Coimbatore 641004, TN, India
anusuya12@gmail.com

Abstract. The electronic health record (EHR) of a patient is a digital record of their health records, progress notes, problems, and medications. As with any online digital format, concerns of breach exist. Similarly, EHR data is susceptible to significant security and privacy concerns these days. This paper has designed a secure decentralized medical blockchain to address privacy and security concerns associated with the exchange of patient health care data across entities. Historically advanced methods for protecting EHRs have often rendered data unavailable to patients. These methods are always attempting to strike a balance between data confidentiality, patient request, and continuous engagement with provider data. The aforementioned difficulties are resolved by blockchain technology, which distributes information in a transactional and decentralised manner. This paper offers a blockchain-based security architecture for efficiently and securely storing and maintaining electronic health records. Moreover, the patients feel insecure about their data being shared so they do not prefer to choose EHR. Maintaining anonymity may facilitate more efficient dialogue between physician and patient, which is critical for ensuring the highest possible quality of treatment. To ensure this, We propose a non-interactive zero-knowledge proof-based authentication method that is lightweight enough to operate on medical health devices with limited resources. The concept of Zero-Knowledge Proof refers to demonstrating anything without disclosing the facts on which the proof is based. It enables physicians, patients, and insurance agents to get medical information efficiently and securely. The goal of this research is to establish if our architecture complies with doctors', patients', and third-party security needs. The simulation results demonstrate that this architecture effectively safeguards EHR data.

Keywords: Blockchain · EHR · Zero knowledge proofs

1 Introduction

A digital health record (EHR) is a computer-based storage system for a patient's health information (history, physical examination, investigations, and treatment). Physicians and hospitals are adopting EHRs due to the many benefits

Supported by organization x.

they provide over paper records. The electronic health record (EHR) is the technology that has the potential to offer the foundation for new functionality and services for patients. They increase health-care access, improve care quality, and lower costs. However, when it comes to EHRs, health care practitioners are faced with ethical difficulties. Patients' autonomy is endangered when health data about them is shared or linked without their agreement. The patient may conceal information out of concern for the system that stores their data's security. As a consequence, the treatment of these patients may be jeopardised. There is a risk that the health data of thousands of patients may be disclosed as a consequence of human mistake or theft.

Additionally, patient access to EHRs is severely limited, with patients unable to share information with physicians or researchers. Conflicts between suppliers, research institutes, and hospitals, to name a few, also obstruct efficient information transfer. EHRs are fragmented rather than synchronised because of a lack of integrated information sharing with management. If a patient is able to keep his or her electronic health records, health departments may profit greatly since it may avoid the need for another doctor to re-diagnose the patient's previous medical history the next time the patient seeks treatment at another facility. By getting blockchain instances from the blockchain, the suggested method accomplishes the goal of increasing cooperation and dependability across all institutions.

To store EHRs, blockchain technology establishes a decentralized healthcare data management ledger [11–15]. When healthcare information is exchanged over peer-to-peer networks, it becomes impervious to modification. To preserve this immutability, the EHR data are connected to one another. Additionally, it protects the privacy of patient EHRs when they are shared and viewed by many users and physicians. The above-mentioned blockchain security architecture protects the integrity, confidentiality, interoperability, authenticity, and accountability of electronic health data shared between two organisations. Individual data is defined as data that will be utilised privately and to which only authorised parties will have access. Zero-knowledge proof is used to ensure privacy. Zero-Knowledge Proof is a relatively new yet very effective idea for establishing trust-based networks. In cryptography, a zero-knowledge proof or protocol is a mechanism for one party (the prover) to show to another party (the verifier) that they know a value x without revealing anything else. The zk-Snark is a kind of extremely secure cryptographic testing that use Zero-Knowledge Proof (ZKP) principles to generate encrypted data that can be readily confirmed without disclosing sensitive information. The inclusion of Zero-Knowledge Proofs in healthcare transactions will assist the sector in optimising its processes and moving toward seamless operations.

The primary benefit is the prevention of assaults from external devices that damage communication between Medical-Healthcare data and their official application. Our strategy is built on a non-interactive zero-knowledge proving method that is both lightweight and resource-efficient.

We suggest a blockchain-based method to solve the problems of storing, managing, and exchanging data. However, the typical blockchain ecosystem does not provide data privacy during transactions or storage. The patient establishes an access policy and distributes the decryption keys to authorised system users. This eliminates the possibility of healthcare professionals collecting or sharing data without permission. As a result, we have a system that ensures patient privacy and is completely self-managed by the patient.

Additionally, we propose a new national healthcare architecture built on the InterPlanetary File System (IPFS) and smart contracts for the storage and administration of electronic health records (EHRs) and other medical records.

2 Related Work

Numerous methods have been suggested to address the issue of interoperability and access control for EHR sharing and insurance claims separately, as well as to address the lack of privacy in EHRs. However, none of the options offered patients with access to their EHR.

Nguyen et al. [1] suggested a novel framework for distributing electronic health records (EHRs) that incorporates blockchain technology and a decentralised interplanetary file system (IPFS) on the mobile cloud. Notably, the authors developed a reliable access control technique based on smart contracts to ensure the secure transfer of electronic health records between different medical providers and patients. The authors concluded that their study demonstrated an efficient method for ensuring the integrity of information transfers on the mobile cloud, including the protection of critical medical data from potential threats.

Sharma et al. [2] has proposed a system for automated insurance claims, interoperability, and data interchange between diverse healthcare providers. They employed a smart card to validate beneficiaries' identities using zero-knowledge proofs and to delegate access to service providers using proxy re-encryption.

Antonio et al. [3] suggested a method for preventing extraneous devices from interfering with communication between medical health devices and their official application. Additionally, they suggested a blockchain-based method to solve the problems of storing, managing, and exchanging data. They proposed a method that utilises Attribute-Based Encryption for data transport, storage, and sharing. A system that ensures privacy and is completely controlled by the patient was suggested. It was constructed using an authentication method based on Non-Interactive Zero-Knowledge Proofs that is lightweight enough to operate on devices with low computing capabilities.

Wanxin Li et al. [4] have presented a new privacy-preserving identity verification method for ridesharing apps by extending zero-knowledge proof (ZKP) and blockchain. They've built a permissioned blockchain network for ZKP identity verification that also functions as an immutable record for ride logs and ZKP data. They develop a protocol for the ZKP module that enables user verification without needing the sharing of any sensitive information. They developed a prototype of the proposed system using the Hyperledger Fabric platform and the Hyperledger Ursa cryptography library, as well as considerable testing.

Adler et al. [5] explain the advantages of a paperless EHR system versus a paper-based one. It also examined the barriers to adoption of a new information technology-based healthcare system and found that a payment system reform was necessary to encourage improved healthcare. We were motivated to use smart cards and zero-knowledge proofs for authentication after reading [6], which used e-passport-based zero-knowledge proofs to establish pseudo-anonymous identities for Sybil-resistant blockchain mining. This approach is ideal for our later-explained verification of identification for healthcare.

Arun et al. [7] They have developed a secure decentralised cloud-based medical blockchain (CMBC) in this study to address privacy and security concerns associated with transferring patient health care data with various medical institutions. Ibrahim et al. [8] developed a blockchain security framework (BSF) for storing and maintaining EHRs efficiently and securely. It provides a secure and efficient method for physicians, patients, and insurance agents to get medical information while preserving the patient's data. Chen et al. [9] presented a blockchain-based concept in which the index for electronic health records is created using complicated logic expressions and maintained on the blockchain. Tanwar et al. [10] offered a strategy that makes use of blockchain technology to address concerns such as the loss of patient privacy, data accessibility, and so on.

3 Blockchain Concepts

A blockchain is a kind of data structure that executes and records transactions in blocks. Each block contains a timestamp and a hash link that connects it to the preceding block. Blocks preserve the integrity of data records and cannot be altered retroactively. It should be seen as a distributed database in which no entity trusts another, and no central point of control exists. Participants may fully depend on one another decentralised. To modify the current block, it is necessary to modify all preceding blocks. As a result, blockchain technology provides a very secure method of transmitting digital assets, money, and contracts without the need of third-party agents. Blockchain serves as a public record of all transactions between participating entities throughout digital activities. Mining is the process of generating a new block. Consensus is reached among all participants to validate each block, ensuring the integrity and trustworthiness of the system. Blockchain technology allows the creation of a decentralised platform amongst participating entities where agreement is reached democratically.

3.1 Ethereum

Ethereum is a public blockchain (all nodes are completely decentralised) in which transactions are sorted (through miners) and updated by individual nodes. Through a method known as Proof of Work, miners earn the privilege to update the blockchain. They enable virtually anybody to engage in the community in nearly any capacity due to their intrinsic architecture, thus boosting adoption

rates. Many of the emerging initiatives want to offer decentralised usefulness to as many people as possible, but they are constrained by problems of scalability and trust.

3.2 Hyperledger Sawtooth

Hyperledger Sawtooth is a business-focused blockchain technology that allows the creation of distributed ledger applications and networks. The design philosophy emphasises the importance of distributed ledgers and the security of smart contracts, especially for business usage. Sawtooth facilitates the creation of blockchain applications by decoupling the core technology from the application domain. Application developers may define the business rules necessary for their application in the language of their choosing, without having to be familiar with the core system's underlying architecture.

3.3 Consensus

Although the primary security feature of blockchain is the use of hashes, attackers may still utilise very costly and very fast machines to recalculate all the hashes, breaching the security layer. To address this issue, the consensus mechanism between blockchain nodes is created. The most well-known consensus mechanisms in blockchain technology are as follows:

- Proof of Work
- Proof of Stake
- Delegated Proof of Stake
- Practical Byzantine Fault Tolerance.

3.4 Consensus Algorithms in Hyperledger Sawtooth

PoET Consensus

Proof of Elapsed Time is a consensus method in the Nakamoto style that is optimised for big networks. PoET is not definitive (can fork).

Sawtooth provides two distinct implementations of PoET consensus:

PoET-SGX implements a leader-election lottery system using a Trusted Execution Environment (TEE), such as Intel® Software Guard Extensions (SGX). PoET-SGX is sometimes referred to as PoET/BFT because to its Byzantine fault tolerance.

On systems lacking a Trusted Execution Environment, the PoET simulator implements the same consensus method. The PoET simulator is sometimes known as PoET/CFT because to the fact that it is crash-tolerant, not Byzantine-tolerant.

PBFT Consensus

Practical Byzantine Fault Tolerance is a finality-assured voting-based consensus method with Byzantine fault tolerance (BFT) (does not fork). Sawtooth PBFT

adds features such as dynamic network membership, frequent view changes, and a block catch-up process to the original PBFT algorithm.

Devmode Consensus

A straightforward random-leader consensus method for testing a transaction processor on a single Sawtooth node. (Devmode is an abbreviation for “developer mode.”) Devmode consensus is not advised for a network with many nodes and should not be utilised in production.

Raft Consensus

A straightforward random-leader consensus method for testing a transaction processor on a single Sawtooth node. (Devmode is an abbreviation for “developer mode.”) Devmode consensus is not advised for a network with many nodes and should not be utilised in production.

3.5 Merkle Tree

A Merkle tree is a kind of data structure that is often used in computer science. Merkle trees are used in bitcoin and other cryptocurrencies to better effectively and securely encode blockchain data. Additionally, they are known as “binary hash trees.” Data verification is critical in a variety of distributed and peer-to-peer systems. This is because the same data is stored in numerous places. Thus, if data is modified in one place, it must be modified elsewhere. Data verification is used to ensure that data is consistent throughout. However, checking the whole of each file anytime a system wishes to validate data is time consuming and computationally costly. As a result, Merkle trees are utilised. We wish to minimise the quantity of data transmitted across a network (such as the Internet). Thus, rather than transmitting a full file across the network, we just send a hash of the file to check for consistency.

4 Zero-Knowledge Proof (ZKP)

Zero-Knowledge Proof (ZKP) is a collection of techniques that enables the validation of a piece of information without exposing the underlying facts. These are probabilistic evaluations, which implies they do not establish anything as conclusively as just disclosing it would. Rather than that, they offer fragments of unconnected evidence that may collect to demonstrate that an assertion’s validity is overwhelmingly likely. The principle behind zero-knowledge proofs is that it is easy to demonstrate possession of a piece of information simply by disclosing it; the challenge is to establish such ownership without releasing the piece of information or any other information. Each transaction is accompanied by a ‘verifier’ and a ‘prover’. In a transaction using ZKPs, the prover tries to prove something to the verifier without disclosing any more information about the item being proved. By supplying the result, the prover establishes that they are capable of computing anything without disclosing the input or the calculation method. Meanwhile, the verifier gains knowledge about the output.

A genuine ZKP must satisfy three criteria:

1. **Completeness:** It should persuade the verifier that the prover is aware of what they claim to be aware of.
2. **Soundness:** If the information is incorrect, it cannot persuade the verifier that the information provided by the prover is genuine.
3. **Zero-knowledge-ness:** It should include no further information for the verifier.

Zero-Knowledge Proof for Privacy Preservation. A blockchain is a collection of records that are jointly maintained by a number of dispersed parties, with each party having a copy of the list. Because blockchains enable all participants to see all transactions, they lack privacy/anonymity. Zero-knowledge proofs enable the posting of private transactions to the blockchain while maintaining their privacy by proving that the transaction was completed properly without disclosing the secret information utilised in the transaction.

5 Proposed Methodology

The above framework illustrates the transactions involving blockchain-based Electronic Health Records. The patient may manage, download, and distribute his or her EHRs autonomously while using a blockchain-based EHR. Five entities comprise the proposed blockchain-based EHR framework: Doctor, Patient, Pharmacy, EHR server, and Insurance Agent. The patient visits the doctor under this method to be treated by doctor. Here, the EHR system server serves as a miner. Whenever an EHR is created for a patient, all the nodes in the blockchain receive it and verify its validity. Once the verification is done it waits inside the memory pool until the miner node takes it and inserts it inside a block. Then the miner starts to verify the information in the block. During verification, a unique hash value is created for that block. Hash is a numeric value that uniquely identifies a block. Once the creation of a block is completed, the miner node distributes it to all the available nodes. Now except for the patient, no one else in the blockchain can view the patient's details. Additionally, the doctor may examine the information of the patients he or she has treated. Because the patient has access to their information, they may share it with the insurance agent.

The insurance agent may see the EHRs of patients who have filed claims on his/her blockchain and, upon approval, can also give the patient with the insurance amount. While buying the medications the patient must disclose the prescription provided by their doctor. The pharmacist must check if the disclosed prescription is valid or not. The pharmacists must give the medicine only if the prescription is valid. If the patient wishes to consult another doctor, they can simply share the details that are stored in the blockchain with them. In blockchain-based EHR, the patient has exclusive access to his EHR information; no one else has access to the information. Additionally, the doctor has access to just the EHRs of patients he has treated. A physician and insurance agents may only see patient blocks that have been given authorization.

In today's healthcare industry, a lot of time-consuming due diligence is done based on a lack of trust. Insurance companies are always wary of fraudulent claims, hence a lot of documentation and details are obtained and analyzed. Doctors need to know more details about their patients such as their insurance status, payment options, etc., during the time of admission. Hence, they do detailed checks. Pharmacists want to verify if the patient was advised to take the medicine or not and then provide them the same. Patients want to ensure if that doctor has a legitimate license with no history of malpractice or any other wrongdoing. To help these entities in verifying their requirements without compromising the privacy of the other entities, the zero-knowledge proof is used. A zero-knowledge protocol is a mechanism for probabilistic verification that involves two parties: a prover and a verifier. The prover is considered to operate in an exponential time domain, while the verifier operates in a linear time domain. The prover's objective is to show that the verifier is aware of a witness, W , without revealing the witness. ZK-snark is the zero-knowledge proof that we will use in our solution. "SNARK stands for Succinct, Non-interactive, Arguments of Knowledge. zk SNARK enables the proof/verification of the correctness of computations without requiring the verifier to run them or divulge any secret information that may have been used in the calculations - the verifier just knows the computation was performed properly." In a healthcare scenario, either of the parties, i.e. patient, doctor, pharmacy, insurance agencies, can take on the role of a verifier, and typically patients and sometimes doctors are the provers. While the ZKP can be applied to any of the transactions involving the above parties, currently the research in the industry is mostly focused on patient privacy rights and ZKP initiatives target more on how much or less information a patient (prover) can share with a verifier before getting the required service based on the assertion of that proof. The above framework includes zero-knowledge proof which preserves the privacy of the patient.

6 Implementation

This paper aims to build a health care system using blockchain technology. We must write smart contracts according to the functions the system is supposed to achieve. First of all, we should design the functions that the system should implement and write corresponding functions in the smart contract for each function to implement it. Ethereum is a decentralised, distributed, and open-source computing tool for developing smart contracts and decentralised applications, or D-Apps. We use Ethereum for writing smart contracts. Primarily, doctors, patients, and pharmacies will be added to the blockchain network. Patients can choose their doctor and send their diagnosis history to the doctor. Once the patient has consulted the doctor, the doctor can update their medical records to the IPFS server.

We combine EHR and blockchain/IPFS architectures to create a distributed database where data may be handled solely by patients and physicians. The IPFS (InterPlanetary File System) provides a decentralized way of storing med-

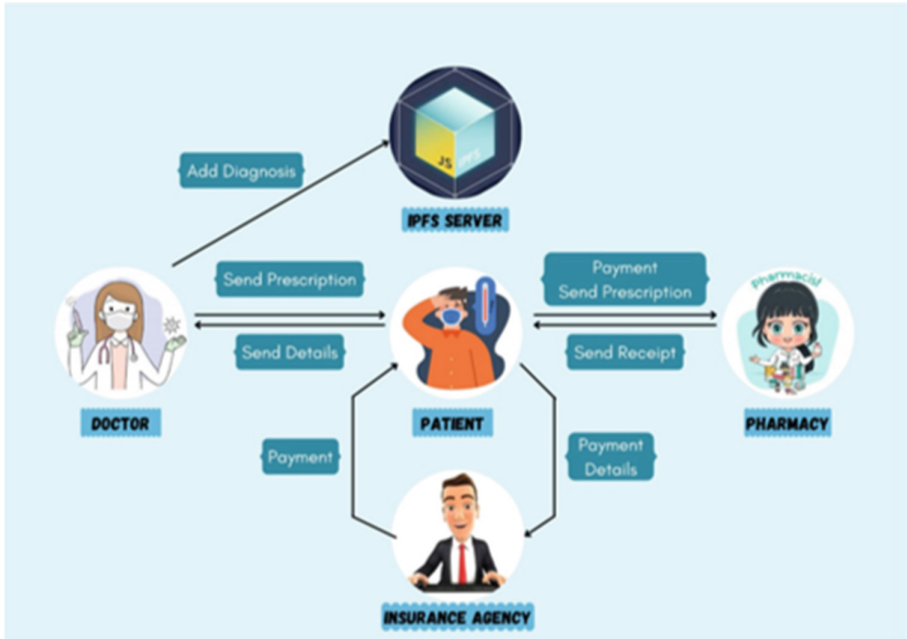


Fig. 1. Design flow of the proposed methodology

ical images, making the records more secure. The required medicines will be prescribed by the doctor and the prescription will also be added. The patient can view their EHRs as well as their prescription. Patients can send the prescription to pharmacies for purchasing the prescribed medications. So far, the contract has realized the basic functions of adding doctors, adding patients, inserting EHRs and prescriptions of the patients, viewing the EHRs by the patients, and sending the patient’s prescription to pharmacies. All the above operations will be permanently recorded in the blockchain to ensure data security and transparency of healthcare records (Fig. 1).

Algorithm Design

- Step 1: The primary function is to add the doctor, the patient, and the pharmacy to the blockchain. The functions `AddDoctor`, `AddPatient`, and `AddPharmacy` help to add these details to the block.
- Step 2: If a patient wants to consult a particular doctor, they will send their details to the doctor through a function ‘`PatientdetailstoDoctor`.’
- Step 3: The doctor analyses the patient and generates an EHR based on the patient’s health condition.
- Step 4: Now the doctor uploads the EHR of the patient to the blockchain using the `setRecord` function and generates the prescription and uploads it to the blockchain with the help of the `setPrescription` function.

- Step 5: To pay the fees, the patient uses a transaction function that accepts doctorId as a parameter.
- Step 6: Now the Patient can access their prescription details and send them to the pharmacy to get the medicines.
- Step 7: Now the pharmacist verifies the prescription and provides the required medicines to the patients.
- Step 8: The patient receives the medicines and pays the pharmacist.

The functions `getPatient` can be used by the doctor to get the details of a particular patient. Similarly, a patient can view a doctor’s details using the `getDoctor` function. Also, a patient can view their details by using the `getRecords` function.

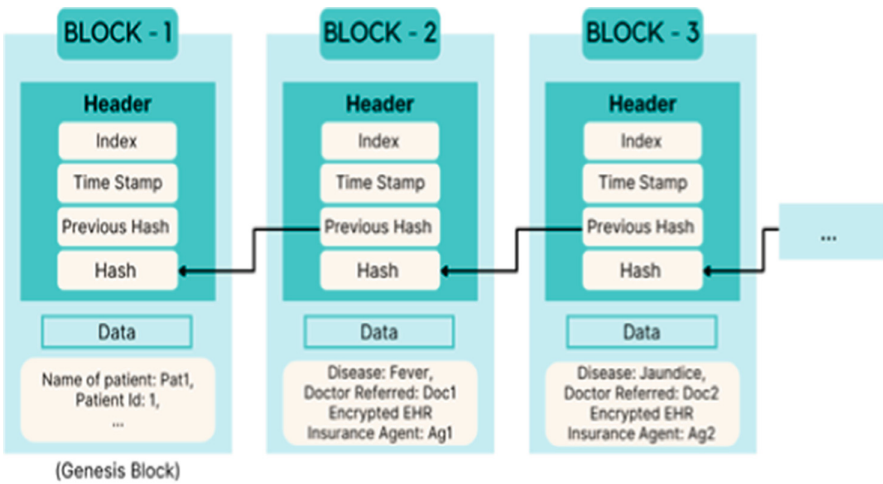


Fig. 2. Blocks storing patient records

Figure 2 depicts the patients’ records stored into the block in the blockchain. These blocks will contain the disease information, the doctor they consulted, EHR data, and insurance agent details related to the patients.

Figure 3 depicts the doctor records stored into the block in the blockchain. These blocks will contain ID, the patient they treated, patient’s disease information, and EHR data related to the patients.

Figure 4 depicts the insurance records stored into the block in the blockchain. These blocks will contain the medical and financial records of the patients.

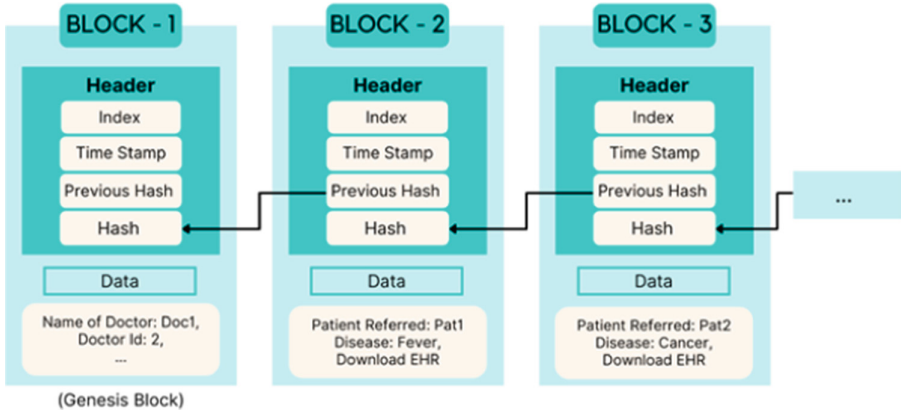


Fig. 3. Blocks storing doctor records

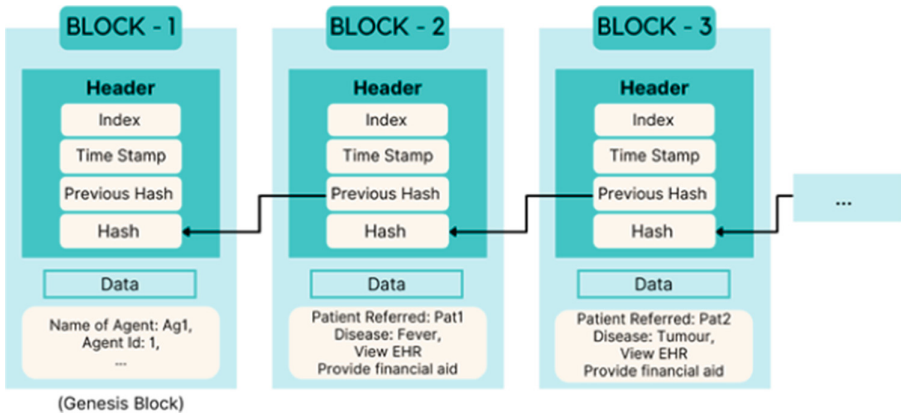


Fig. 4. Blocks storing insurance records

6.1 Zero-Knowledge Proof Algorithm

To achieve privacy in our blockchain network we use Zero-Knowledge Proof.

- **Input:** documents shared by the prover.
- **Output:** true for a valid proof and false for an invalid proof
- **Step 1:** Verifier compiles and runs the zok file, creating proving and verification keys. “proving.key” file is sent to the prover.
- **Step 2:** The prover compiles the zok file independently to make sure it doesn’t disclose information they want to keep secret.
- **Step 3:** The prover creates a witness file with the values of all the parameters in the program. Using this witness, the pharmacy’s proving.key and the compiled program generate the actual proof. It is created in the file “proof.json”.
- **Step 4:** prover shares the “proof.json” file with the verifier. Verifier can verify the file.

- **Step 5:** So far, the prover and the verifier have validated the prescription between themselves. However, it is often useful to have the values published on the blockchain. To do this, the verifier creates a solidity program, “verifier.sol”.
- **Step 6:** To verify, the verifier file is compiled, deployed, and checked for the result to be true.

Let the above algorithm be called as a function named `zkp` which takes verifier, prover, and `zokFile` as the parameters. `ZokFile` is a `zok` extension file that contains the logic for verifying the documentation provided by the prover.

`zkp(verifier, prover, zokFile)` The above algorithm is implemented for

- Patients want to ensure that doctors have a legitimate license with no history of malpractices. Here the patient is the verifier and the doctor is the prover. `zkp(patient, doctor, zokFile)`
- Doctors need to obtain a lot of documentation from patients for verifying insurance. Here the doctor is the verifier and the patient is the prover. `zkp(doctor, patient, zokFile)`
- Pharmacists have to verify that the patients are indeed advised by a valid doctor to take the medicines. Here pharmacists act as a verifier and patients act as a prover. `zkp(pharmacy, patient, zokFile)`.

7 Results and Discussions

We provide an experimental study of a solidity smart contract-based authentication method. For building and deploying our code, we utilised a 8GB RAM i5 CPU. As advised by the instructions, the Zokrates library was utilised in a docker container, and the health care data was stored via IPFS. To determine the scalability of our authentication strategy, we used Ganache to simulate a blockchain without requiring long block mining. A hyperledger caliper is used for the performance analysis of the smart contract implementation in hyperledger sawtooth and ethereum.

Ethereum. The below tabulation in Fig. 5 shows the performance analysis of deploying smart contracts in ethereum.

Graphical representation of performance analysis is shown in Fig. 6

Name	Success	Fail	Send Rate (TPS)	Max Latency (s)	Min Latency (s)	Avg Latency (s)	Throughput (TPS)
Fixed TxnCount-TxnPerBatch (16)	16	0	18.5	4	3.13	3.57	4
Fixed TxnCount-TxnPerBatch (120)	120	0	170.5	8.45	3.46	5.89	13.2
Fixed TxnCount-TxnPerBatch (225)	225	0	271.1	18.08	3.69	9.27	11.9
Fixed TxnCount-TxnPerBatch (560)	560	0	530.3	38.01	3.96	20.18	14.3

Fig. 5. Performance analysis of ethereum

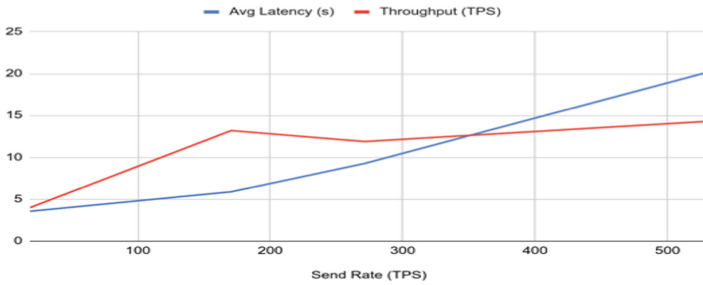


Fig. 6. Graphical representation of performance analysis of ethereum

7.1 Hyperledger Sawtooth

The below tabulation in shows the performance analysis of deploying smart contracts in hyperledger sawtooth (Fig. 7).

Name	Success	Fail	Send Rate (TPS)	Max Latency (s)	Min Latency (s)	Avg Latency (s)	Throughput (TPS)
Fixed TxnCount-TxnPerBatch (10)	16	0	19.8	0.49	0.45	0.47	12.3
Fixed TxnCount-TxnPerBatch (100)	120	0	198.3	2.84	1.72	2.28	34.8
Fixed TxnCount-TxnPerBatch (200)	225	0	351	6.14	2.16	4.12	33.2
Fixed TxnCount-TxnPerBatch (500)	560	0	598.3	16.16	2.39	8.94	32.8

Fig. 7. Performance analysis of hyperledger sawtooth

Performance analysis is shown in Fig. 8

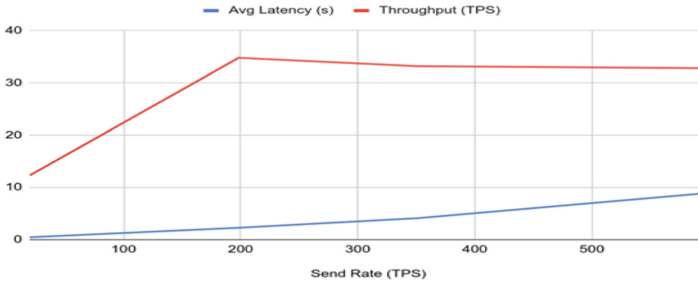


Fig. 8. Graphical representation of performance analysis of hyperledger sawtooth

From the performance analysis of ethereum and hyperledger sawtooth, we can observe that for a sending 560 successful transactions, in ethereum the TPS (Transactions Per Second) is 14.3 whereas in hyperledger sawtooth, the TPS is 32.8. Hyperledger sawtooth is faster than ethereum. We propose a decentralised healthcare system in this article to safeguard data integrity and privacy in a scenario involving several blockchain-based linked networks. Our technology integrates zero-knowledge proof into the healthcare system in a novel way, allowing us to validate a doctor’s licence, a legitimate prescription, and a patient’s insurance information without disclosing any sensitive information.

8 Conclusion

This article offers a security architecture utilizing blockchain for effectively and securely storing electronic health data. We highlighted the impediments to implementing a ‘conventional’ Blockchain-based EHR sharing architecture. This methodology ensures that patients have unrestricted access to EHRs while also protecting patients’ privacy and maintaining EHR consistency. In our system, we employed zero-knowledge proofs for authentication, which allowed us to distribute EHR access swiftly and securely while respecting patient privacy. The patient may manage, download, and distribute his or her EHRs autonomously utilising this blockchain architecture. The testing findings demonstrate that our blockchain enables users to share data securely. Most significantly, the blockchain security framework access control system is capable of safeguarding critical electronic health records from external assaults. Investigating these advantages of our approach has the potential to revolutionise healthcare and accomplish the goal of privacy preserving EHRs. This research used the blockchain security architecture only in the healthcare sector. We want to apply this paradigm in the future to a variety of areas, including supply chain management, IoT, agriculture, finance, smart grid, education, logistics and finance.

References

1. Nguyen, D.C., Pathirana, P.N., Ding, M., Seneviratne, A.: Blockchain for secure EHR sharing of mobile cloud-based e-health systems. *IEEE Access* **7**, 66792–66806 (2019)
2. Sharma, B., Halder, R., Singh, J.: Blockchain-based interoperable healthcare using zero-knowledge proofs and proxy re-encryption. In: 2020 International Conference on COMMunication Systems & NETworkS (COMSNETS), pp. 1–6. IEEE, January 2020
3. Tomaz, A.E.B., Do Nascimento, J.C., Hafid, A.S., De Souza, J.N.: Preserving privacy in mobile health systems using non-interactive zero-knowledge proof and blockchain. *IEEE Access* **8**, 204441–204458 (2020)
4. Li, W., Meese, C., Guo, H., Nejad, M.: Blockchain-enabled identity verification for safe ridesharing leveraging zero-knowledge proof. In: 2020 3rd International Conference on Hot Information-Centric Networking (HotICN), pp. 18–24. IEEE, December 2020
5. Adler-Milstein, J., Bates, D.W.: Paperless healthcare: progress and challenges of an it-enabled healthcare system. *Bus. Horiz.* **53**(2), 119–130 (2010)
6. Kalaipriya, R., Devadharshini, S., Rajmohan, R., Pavithra, M., Ananthkumar, T.: Certain investigations on leveraging blockchain technology for developing electronic health records. In: 2020 International Conference on System, Computation, Automation, and Networking (ICSCAN), pp. 1–5. IEEE, July 2020
7. Arunkumar, B., Kousalya, G.: Blockchain-based decentralized and secure lightweight E-health system for electronic health records. In: Thampi, S.M., et al. (eds.) *Intelligent Systems, Technologies and Applications*. AISC, vol. 1148, pp. 273–289. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-3914-5_21
8. Abunadi, I., Kumar, R.L.: BSF-EHR: blockchain security framework for electronic health records of patients. *Sensors* **21**(8), 2865 (2021)
9. Chen, L., Lee, W.K., Chang, C.C., Choo, K.K.R., Zhang, N.: Blockchain-based searchable encryption for electronic health record sharing. *Future Gener. Comput. Syst.* **95**, 420–429 (2019)
10. Tanwar, S., Parekh, K., Evans, R.: Blockchain-based electronic healthcare record system for healthcare 4.0 applications. *J. Inf. Secur. Appl.* **50**, 102407 (2020)
11. Vora, J., et al.: BHEEM: a blockchain-based framework for securing electronic health records. In: 2018 IEEE Globecom Workshops (GC Wkshps), pp. 1–6. IEEE, December 2018
12. Yang, G., Li, C.: A design of blockchain-based architecture for the security of electronic health record (EHR) systems. In: 2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom), pp. 261–265. IEEE, December 2018
13. Shahnaz, A., Qamar, U., Khalid, A.: Using blockchain for electronic health records. *IEEE Access* **7**, 147782–147795 (2019)
14. Ivan, D.: Moving toward a blockchain-based method for the secure storage of patient records. In: *ONC/NIST Use of Blockchain for Healthcare and Research Workshop*. ONC/NIST, Gaithersburg, Maryland, United States, pp. 1–11. Sn, August 2016
15. Tang, F., Ma, S., Xiang, Y., Lin, C.: An efficient authentication scheme for blockchain-based electronic health records. *IEEE Access* **7**, 41678–41689 (2019)