BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **22**, No 4
Sofia • 2022 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2022-0043

A Decentralized Medical Network for Maintaining Patient Records Using Blockchain Technology

M. Sumathi[1], *S. P. Raja*[2], *N. Vijayaraj* [3], *M. Rajkamal*[4]

[1]*School of Computing, SASTRA Deemed University, Thanjavur, Tamilnadu, India*
[2]*School of Computer Science and Engineering, Vellore Institute of Technology, Vellore-632 014, Tamil Nadu, India*
[3]*Department of Computer Science and Engineering, Vel Tech RangarajanDr.Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, Tamil Nadu, India*
[4]*Application Developer, IBM, Bangalore, Karnataka, India*
E-mails: sumathishanjai.nitt@gmail.com avemariaraja@gmail.com

***Abstract*:** *Personal Medical Records* (*PMR*) *manage an individual's medical information in digital form and allow patients to view their medical information and doctors to diagnose diseases. Today's institution-dependent centralized storage, fails to give trustworthy, secure, reliable, and traceable patient controls. This leads to a serious disadvantage in diagnosing and preventing diseases. The proposed blockchain technique forms a secured network between doctors of the same specialization for gathering opinions on a particular diagnosis by sharing the PMR with consent to provide better care to patients. To finalize the disease prediction, members can approve the diagnosis. The smart contract access control allows doctors to view and access the PMR. The scalability issue is resolved by the Huffman code data compression technique, and security of the PMR is achieved by an advanced encryption standard. The proposed techniques' requirements, latency time, compression ratio and security analysis have been compared with existing techniques.*

***Keywords*:** *Healthcare, Blockchain, Smart contract, Ethereum, Personal Medical Record, Huffman code, Advanced Encryption Standard.*

1. Introduction

Blockchain (B_C) is a decentralized database, and blocks are shared by all nodes within the B_C network. B_C stores data in a digital form like other databases. Blocks act as the fundamental building blocks of the B_C network. The storage capacity of each block is restricted to 1MB. If the blocks are filled, they are sealed and connected to the previous block. Similarly, every block is connected to the previous block and forms a chain, which is known as B_C. Once the B_C is created, the data stored in the blocks is irreversible [1]. The B_C can hold a variety of data, like medical, agriculture, transactions, etc. The secure and verifiable B_C networks are created by the Ethereum

platform and specialized language known as Solidity. The major advantage of B_C technology is that a central authority cannot control the network; instead, transactions are able to be created by anyone in the network [2]. Conventionally, in the medical industry, patient information is stored on a hospital server and works on a client-server model. The patients can upload their details onto the server, and can later request access to view their details via any device connected to the server. This could be done by a simple file transfer protocol, or the server could also host a website, which then could be used by the end-users. This system poses a single point of failure and can be manipulated illegally by anyone with malicious intentions. This technique fails to provide reliable and secure control over the patient's data, which is a serious threat to authenticity and accuracy. Changes made to the data could not be traced back, making it difficult for the users to trust the system [3].

Instead of using non-trustable storage for Personal Medical Records (PMR), the B_C network provides a more secure and decentralized system for handling the details, which makes the architecture more transparent. In this system, there is no scope for a single point of failure. Since everyone in the network is responsible for handling the data, just like any other peer-to-peer system. Not only are the patient details stored making use of B_C technology, but a secure network for doctors of a particular specialization for gathering opinions on a particular diagnosis by sharing the medical record with consent from the patient is also present in this architecture. This system allows the doctor to reach a consensus for a particular diagnosis by viewing the total up votes given by the other doctors in the network. The specialized doctor's network is helpful in making better decisions about the patient's treatment instead of single doctor-based treatment [4].

Similarly, the B_C technique is helpful in sharing the PMR between physicians, hospitals, pharmacies, and diagnostic laboratories. Likewise, in security aspects, B_C provides authentication, versatility, accountability, and interconnection in an efficient way for data access. This security aspect avoids security threats in the healthcare sector. The major advantages of B_C in the healthcare sector are that it provide resource access at all times to authorized users without delay (providing quicker treatment in emergency cases) and it provides controllable access (authorization verified by one third of the verifier). The B_C provides visibility, data privacy and identity privacy of individuals. The major features of the B_C in healthcare are: efficient PMR data management, point of care genomics management, quicker disease tracking and outbreaks, secured managing of PMR, inter-operable PMR records, protection of healthcare data, point of care genomics management, and safeguarding genomics, etc. As a result, the B_C has been adopted in healthcare sectors [17].

The remainder of this article is structured as follows: Section 2 is focused on a review on the literature on B_C-based PMR storage. The proposed technique along with the necessary algorithms and equations is discussed in Section 3. In Section 4, the experimental results of the proposed work are compared with the results of previous work. In Section 5, the performance analysis of the proposed system is discussed in different aspects. The proposed technique is concluded in Section 6 with the future enhancements.

2. Literature review

Mohammad Moussa Madine [5] have used Solidity to create Smart Contract (S_C) in an Ethereum-based B_C for storing PMR. In this technique, patients have control over their PMR. The security of the records is solidified using Oracle. However, the network created for storing the PMR cannot be shared with other doctors [5]. Raifa Akkoui, Hei and Cheng [6] have proposed a method using IPFS to have a decentralized storage of PMR, which has the advantage of having more storage, despite block size limitations in BC. This network is also observed to have high throughput and low system resource utilization. But, the data is anonymised by omitting certain information protections [6]. Abdullah Al Omar [7] discuss a healthcare system for smart cities. The PMR has been shielded with more cryptographic tools, and the S_C is written using solidity. Since the S_C has not been implemented on the central network, there is a risk of a possible failure of the central point [7]. Yan Zhuang et al. [8] propose patients control over the PMR based on S_C. By using S_C, the patient can easily control who has access to their PMR and other related details. This technique is limited to only one private network, which leads to scalability issues [8].

Ilhaam, Jayaraman and Debe [9] discusses how the patients make use of the automation of the GPO contract process that is present in the S_C being deployed. The major drawback is that other stakeholders who make use of this network do not have automated processes, and everything has to be handled manually [9]. Zhang et al. [10] have proposed the patient controllable PMR management based on B_C. The PMR process has been improved in several areas, including telehealth and medicine, patient health insurance claims, patient controlled PMR, and so on. The secure PMR record maintenance and transfer is discussed in detail [10]. Griggo et al. [11] use the private B_C to provide privacy to patients' PMR. Due to higher sensitivity level, the private B_C has been used to store PMR. The private B_C allows only the registered and authorized users to access the PMR with the acknowledgement of the members that are involved in the network [11]. Sharma et al. [12] analyse the B_C-based PMR data sharing between the participating patients. Precision healthcare management is a process for improving the security and privacy levels of precision along with the trust levels between patients and doctors. The trust level is higher than the predefined threshold value; the specific request is able to access the PMR, otherwise unable to access the PMR [12].

Pongnumkal, Siripanpornchana and Thajchayapong [13] propose the Hyper-ledger Fabric technique for constructing the B_C network. This technique provides a better B_C network than the other B_C network. Almost more than 10,000 transactions have been taken for the process. The cost and traffic of network transactions are high for more than 10000 transactions. The fabric network works well for less than 10,000 transactions, but not for more than 10,000 transactions [13]. Nihar et al. [14] have proposed the B_C-based PMR management system. The hyper ledger-based S_C has been proposed for providing better PMR management than centralized data management. The performance of the multi-hosted system analysis has outperformed that of the Gossip-based performance. This multi-hosted technique uses low utilization of resources and latency to achieve high throughput in the PMR

process [14]. V a h i n y S h a r m a et al. [15] has analysed the B_C-based PMR secure processing system. The B_C-based secure data storage and sharing is proposed for improving the security and integrity of PMR. The B_C-based record maintenance system avoids vulnerability, single point failures, privacy risks, and data dispersion issues [15].

S a l a h et al. [19] have discussed the B_C concepts. The B_C technique administrative costs are reduced through public key cryptography, peer-to-peer networking, and consensus mechanisms. The public B_C is accessible and transparent to all. Instead, the private B_C technique is used within a single organization and gives complete control over the stored data. The B_C consortium combines the advantages of public and private B_C. To streamline communication the consortium is controlled by few organizations [19]. I b r a r Y a q o o b et al. [20] have examined the B_C technique in the healthcare sector from various perspectives. To improve secrecy, the PMR stored in B_C is divided into multiple parts and stored separately. Similarly, the PMR has been stored in the blocks using a standardized code. The B_C storage safeguards the data against natural disasters as well as data mishandling or theft. The PMR storage in B_C allows faster access and global traceability. In auditability, a public ledger makes it simple to verify data access, and global storage eliminates the need for redundant data storage [20].

Based on the above analysis, the following merits are identified in the existing work: the S_C is deployed on B_C. So without single point of failure, the decentralized and more secure PMR storage is achieved through B_C. The PMRs are uploaded using the internet protocol file sharing system, which is decentralized and helps in overcoming the size limit of a single block in the network. Patients belonging to a network can trace their PMR at any stage, making it more transparent and traceable. All the previous interactions with the network by the user can be viewed by anyone who is a part of the network.

The limitations of the existing techniques are: the number of doctors who can join the network is limited by the network limit that has been set by the administrator of the particular network. The manager/administrator of the network cannot change the network value to a lower value than the previous one. That is, the network limit can only be increased, and cannot decrease the count of the doctor from the network by an administrator. A patient belonging to a network can upload only one medical report file per patient, thus any changes made in the future cannot be stored on the IPFS file storage system. Once a record has been uploaded to the network, it cannot be deleted or modified. To overcome the aforementioned demerits, data compression and encryption based PMR storage in B_C is proposed in this work.

The contributions of the proposed work are as follows,

- The doctors will be dynamically added or removed from the B_C network via S_C.
- The most recently changed PMR is going to be stored in a block. The separate document storage of each change aids in the efficient tracking of the patients health condition, i.e., helps to determine whether the patient's health condition has improved or not.

- After one-third of the member's confirmations, the PMR block will be constructed and added to B_C.
- To overcome the storage size issue, the PMR is going to be compressed by compression technique and stored in the block.

3. Proposed work

Two contracts comprise the overall process of PMR, such as doctor network and patient file. Fig. 1 shows the proposed system architecture. In a doctor network, the S_C allows doctors to create their own network, based on the specialization. The doctor can set the network limit in a dynamic manner. That is, able to control the number of doctors who can be in the network at a particular time. Once the limit has been set for a specific process, it cannot be reduced. This helps to take better and more accurate decisions in emergency cases. Frequent update in a network leads to poor decision making and it is difficult to manage the update. The dashboard displays the number of doctors currently working on the specific case, and it also displays the number of patient reports published under it. A network contract allows doctors to register with a specific network until the network limit is reached. This S_C allows doctors to initiate diagnosis verification by creating a patient report with all the necessary patient details. If a patient has granted access to another doctor to add or view their report, the authorized doctor can verify another doctor's diagnosis report. The doctor who initiated report verification can finalize it after getting an approval from one third of doctors in the network.

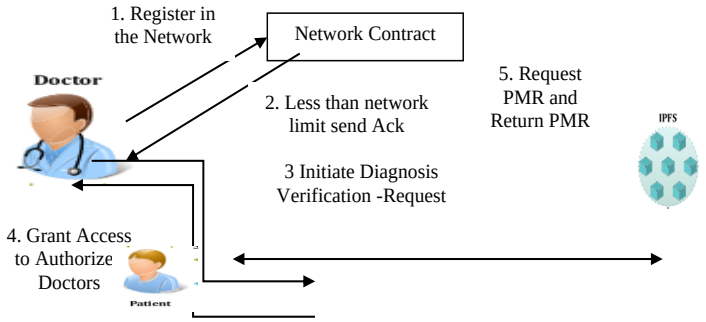

Fig. 1. Holistic view of the proposed system architecture

The central contract is responsible for administering the entire network. The Data owner (D_O) creates the S_C and deploys it onto the test or main network. The ownership can be transferred to any other person by the wish of the previous D_O. The D_O's and admins can add and remove admins responsible for handling a branch of the network they have created. The admin can add new hospitals based on request from the patients or doctors if they are interested in a decentralized storage of patients' records. The doctors in a network can view the PMR details, if the patient has provided access rights to them. By allowing the doctor's address, the doctors view or access the PMR stored in IPFS. If their address (64 bit unique address) has been

provided with permission, the doctor could also add PMR in their respective allotted space.

The patients belonging to a particular doctor network can sign-up using meta-mask if they are using the network for the first time, or they can also login through user ID if they have already registered. Patients can grant and revoke access rights to the doctor. They can also add more than one doctor's address to grant or revoke access to their report. Similarly, patients can also add their own PMR. If the doctor has already added a PMR, the patient can view and update their profile if any discrepancies are present in PMR.

The major benefit of the B_C is that it stores the block information in the form of a hash value. The well-known and appropriate algorithm for implementing B_C technology is the Secured Hash Algorithm (SHA)-256. In the proposed work, the user information is stored in the form Hash value (H_V). The next equation is used to generate the H_V of each block:

$$H(PMR) = PB_{HV} + C(PMR) + N + T_S + P_{ID}. \quad (1)$$

Each block $H(PMR)$ consists of the Previous Block Hash Value (PB_{HV}), encrypted PMR $C(PMR)$, Nonce (N), Timestamp (T_S) and Patient ID (P_{ID}). By using $PB_{HV,}$ the immutability of the block is maintained. When an unauthorized user tries to change the block information, the PB_{HV} also needs to change. This is an impossible task. Thus, data integrity is maintained in the B_C through PB_{HV}. The N and T_S values ensure the time period of block generation, access, and transfer. By using block information, the access history of PMR can be identified in an accurate manner.

The PMR consists of highly sensitive health information about an individual. Hence, protecting the sensitive information is an essential task of PMR record maintenance. Thus, the Advanced Encryption Standard (AES) technique is used to convert the plaintext record to a ciphertext record in the proposed system. The AES process consists of four steps such as sub-bytes, shifting rows, mixing columns and adding round key. In a sub-byte process, every byte of the input matrix is substituted with a value that is neither the same as nor its complement by using an S-box lookup table. The result of s-box is then fed to the shift rows step. In a shift rows, every row in the input array after being substituted in the previous steps is shifted to the left in a circular manner. Afterwards, the matrix is generated based on matrix multiplication with a specific matrix, thereby changing the position of each byte in the original matrix. Finally, in adding round keys, the output of the previous stage is XOR-ed with the appropriate round key. Decryption can easily be performed by reversing the steps used in encryption, but the mixed column should not be reversed in the first step since it is skipped in the last step of encryption. Algorithm 1 shows the working procedure of the proposed system.

Algorithm 1. B_C-Based PMR maintenance

Input: PMR, AES Key, SHA-256, Doctor Request

Output: C(PMR), H(PMR), Response to Doctor

Procedure:

Step 1. For all PMR do

Generate secret key K of D_O using elliptic curve cryptographic algorithm

$$y^2 = x^3 + ax + b$$

$\mathrm{C(PMR)} = \mathrm{Eny(PMR} \oplus K)$

Step 2. For each C(PMR) do

Generate hash code of PMR

$\mathrm{H(PMR)} = \mathrm{PB_{HV}} + \mathrm{C(PMR)} + \mathrm{N} + \mathrm{TS} + \mathrm{P_{ID}}$

Step 3. Store H(PMR) in IFPS

Step 4. New doctor register in network contract and get the ID

Step 5. Doctor Send the ID to D_O

Step 6. D_O verify the authorization and Ack to get the PMR from IFPS

Step 7. Doctor send the Ack to IFPS

Step 8. Doctor verify the threshold 'th' of the requested block

Step 9. If (block verification value $\geq$ th)

Block accessed

Else

Block rejected

The block size limitation overcomes by, Compression Technique (C_T). The C_T helps to reduce transmission latency, storage capacity, and improve energy efficiency. There are two types of C_T that are lossless and lossy compression. In the medical field, every piece of measured data is crucial in decision making. As a result, lossless C_T is preferable than lossy C_T. In comparison to other C_T, the Huffman coding Technique (HC_T) is a popular lossless data C_T. The maximum frequency data has a smaller code size than the minimum frequency data in the HC_T. This property significantly reduces data size. Each PMR is compressed and stored in a separate block. The compressed code size is calculated using the equation

(2) $$\mathrm{Cmp}\big(\mathrm{H(PMR)}\big) = \sum_{i=1}^{n} \mathrm{Wc}_i * \mathrm{len}(\mathrm{W}_i),$$

where Wc_i – represents the number of Words in a PMR, and len(W_i) represents the number of characters in each word. Now, the Compressed Data (C_D) is stored in blocks. At access time, the C_D is decoded and then the decryption process is applied to get the original data. To overcome the scalability issues, the PMR record with sensitive information has been stored in a permissioned off-chain mode and the remaining information has been stored in the cloud storage. This storage partition technique provides scalability and extendibility to PMR storage. Attribute-Based Access Control (AB-A_C) is used to achieve A_C. To assign and verify the A_C, the user groups are divided into clusters based on the type of user, such as D_O, specialization-based doctors' groups, and so on. The access rights are determined by the user's ID, role and membership type. The user type determines the access limit for each group. The AB-A_C is adaptable add and remove users dynamically.

Before being stored PMR in a block, the requested block verified by the 'th' number of doctors. If the block is verified by more than the thresholded doctors, the prediction accuracy is high and it is helpful for quicker decision making by the new doctor. Thus, the fast decision making process can be performed in the proposed system in a secured and scalable manner.

4. Experimental results

The proposed system has been implemented using the solidity compiler (Remix), the next.js framework, visual studio code and the meta-mask browser support extension. When the report was uploaded to the network, patient's PMR has been added into the IPFS file management system. The PMR being fed was fetched from the patients via PDF format. PMR contain medical image or scan sheets. Hence, the PDF file occupied more space in blocks. The block size has a limit of 1MB. However, since a decentralized storage system is used, users need not store the files onto the blocks, but instead store them on the IPFS file management system, and then fetch the file hash and upload them instead on the B_C. By the IPFS and C_T, the storage space limitation of B_C has been resolved in the proposed technique.

The Home page is used for navigating to the Doctor Network page for creating new networks, or to view the patient's records, already established on a previously created network. After that, three features are available on the patient record homepage such as admin login, doctor login and patient login or creating a new patient account. The admin dashboard is used by the network administrator to create, remove, or change the manager of the particular network. In any organization, role change is a common factor. Thus, to create, remove, or change of admin is an essential task. In the proposed work, the admin is added to the network to manage the entire data processing (add doctors, patients, block generation and access, block storage, etc.).

The major benefit of the proposed work is that everything is stored in the form of a H_V. Hence, the block information cannot be viewed or changed by an admin and others. The proposed work has been implemented in the permissioned B_C. Hence, only registered users can participate in the network. The addition and removal of the node is done by admin of the network. The node creation is dependent on the role not based on individual person. If anyone is remove from the organization, other person will join and act in this role. So the network infrastructure will not be affected by the change of role.

If the patient has provided appropriate permission, doctors are able to use the doctor dashboard to view their own information, add a new patient record, and view the patient record. Only registered and authorized doctors can add and view patient data. Others are unable to do so. If the doctor is a newly registered physician, he/she needs to obtain access rights from the network administrator and proceed with the process. If the doctors are a removed, the admin will remove their role and they will be unable to access the block using their existing key. The patient dashboard is used by the patient to access a variety of features such as adding a record, giving or removing permission for doctors to view their information, uploading reports, and so on. The new patient must first register in the network in order to access, monitor, and control their documents. Patients who are not authorized or registered cannot store or access the PMR on the network.

The network homepage is used to view previously created networks and to create new ones. The network addresses are represented as hash codes, which ensures the network and patient records' integrity. A network can be created by providing a proper name for the network. Since the changes made are reflected on the B_C, a

typical interaction takes a few seconds. Once a network is operational, registered doctors can view network details. Only the network administrator has the ability to change the network limit and add doctors.

5. Performance analysis

The performance of the proposed system is analysed in two different aspects, as the computational effort required for constructing the blocks and latency time.

Computational Effort Requirement (Gas requirement). The computational effort requirement is analysed based on the gas requirement to execute the task of varying input sizes. Fig. 2 shows the computational time requirement of the proposed system analysis in milliseconds (ms) with an input size of KB's. Based on the analysis, input size increases; gas requirements also increase. Thus, gas requirements and input size are direct proposals to each other.

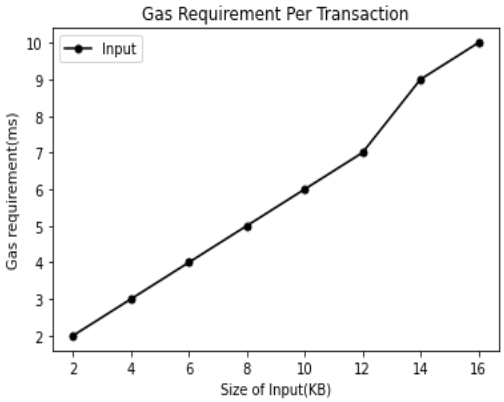

Fig. 2. Computational effort requirement analysis

Latency Time based on input size. Latency time is the time duration between the transaction initialization and completion so this is the total time required for completing the block generation. Fig. 3 shows the latency time analysis of the proposed system.

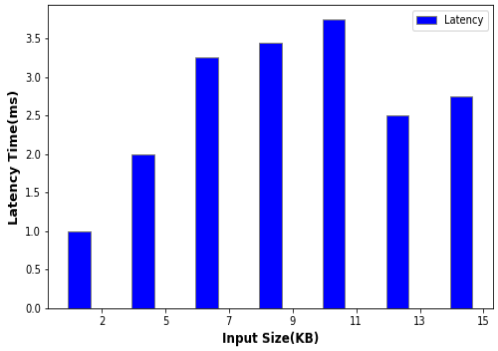

Fig. 3. Latency time analysis – input size

The input size is given in KB and the time is calculated in milliseconds. Due to test environment inconsistency, the latency time varies for each size of input. The latency time rising is dependent on the block size such as higher resources or bandwidth propagate to larger block size and leads to network congestion. This varying block size and network congestion lead to latency variations. Likewise, depending on number of nodes the latency time varies. So as lesser number of nodes takes lesser time to confirm, the data and larger number of nodes take more time to confirm the data. This can also leads to irregular increase of the latency time.

Utilization of storage space. The main issue with the B_C technique is storing the limited size data in the block. As a result, C_D have been stored in the blocks instead of actual data. C_D storage requires less storage space (lesser than 65%) and transfer time than un-C_D storage. Table 1 compares the storage requirements of C_D and un-C_D storage. The next equation is used to calculate the required space:

$$\text{space}_{\text{req}} = \frac{\text{size (before compression}-\text{after compression)}}{\text{size before compression}} \times 100\ \%. \quad (3)$$

Table 1. Storage space requirement comparison

Uncompressed storage (KB)	Proposed compressed storage (KB)
22.09	13.82
44.35	27.35
15.38	8.96
11.22	7.58
78.14	45.36
39.89	23.29
115.20	73.05

Scalability. The efficient storage space utilization leads to resolving of scalability issues in the B_C technique. Nearly 45% of storage space is reduced in the proposed technique. This storage space reduction leads to store large number of PMR in a B_C network. Hence, the scalability issues are resolved in the proposed technique.

Latency Time based on Number of users: The number of users involved in the process also affects latency time. When the number of users increases, so does the latency time. This work has generated 500, 1000, 1500, and 2000 user groups. The latency time is calculated as a range of minimum, average, and maximum values. Fig. 4 depicts a comparison of latency time analysis for various user groups.

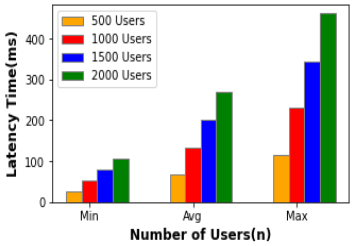

Fig. 4. Latency time analysis – number of users

Comparison of proposed and existing scheme. In Table 2, the proposed system's key performance is compared to existing schemes such as Hierarchical Schemes (HS), Pre-shared Key Schemes (PKS), Mathematical Framework (MF),

Key Pool Framework (KPF), and Public Key Centres (PKC). The proposed scheme has no centralized components. Auditability is performed in a dynamic manner based on consensus. To reduce update complexity, data and access history updates are performed dynamically in the compressed storage representation. The B_C technique is entirely decentralized, and the compressed data is distributed among the members of the proposed technique. This reduces the amount of communication and computation required. As a result, the proposed technique outperforms the existing system, as demonstrated. The parameters in Table 2 indicated by: U – Untrusted; H – High; L – Low; M – Medium; T – Trusted; CA – Certificate Authority; C – Consensus.

Table 2. Key performance analysis [20]

Parameters	HS	PKS	MF	KPF	PKC	Proposed
Auditability	U	U	U	U	T-CA	T-C
Scalability	L	L	H	H	M	H
Extensibility	L	L	M	M	H	H
Communication and Computation overhead	H	L	L	L	H	L
Central point of overhead	M	M	M	M	H	N

6. Security analysis

Data security analysis is an essential task for data confidentiality, integrity and availability. Confidentiality ensures data secrecy; integrity ensures data consistency; and availability ensures data availability to authorized users.

1. Confidentiality. The B_C-based decentralized system has been implemented in a permissioned B_C network and is created between the authorized users only. Hence, the data confidentiality of patient PMR is maintained without compromising the privacy of patients. Similarly, to provide higher confidentiality to PMR, the AES encryption technique is applied to PMR before storing in the B_C. Thus, confidentiality is maintained in the proposed system. Table 3 shows the comparison of proposed B_C and existing cloud-based PMR maintenance.

Table 3. Security analysis of B_C-based and Cloud-based PMR storage

Parameters	Cloud-based PMR management [15, 16]	Proposed BC-based Management
Decentralized Storage	No	Yes
Privacy	Yes	Yes
Decentralized Execution	No	Yes
Trustfulness	Partial	Yes
Immutability	Partial	Yes
Patient-centric Control	Partial	Yes
Provenance	Partial	Yes
User Centric and Authentication	No	Yes

2. Internal and External Attacks. The internal and external attacks are avoided in the proposed system in different levels. Tthe decentralized storage system, eliminates the single point of failure issue (avoids downtime attacks), the requester access permission is verified by multiple members that are involved in the network (avoids malicious access), and having the ability to create blocks based on multiple

member's confirmed (avoids false data creation). Thus, internal and external attacks are eliminated in the proposed work.

3. Integrity and availability analysis. The major feature of the B_C technique is to store the information in the form of a hash code. The same hash generation of the modified data is an impossible task. Similarly, in the B_C technique, modifying one specific block and generating an equal code to the existing hash value is an impossible task. Because in B_C each block hash value depends on PB_{HV}. To change one PB_{HV} is a complex process. Hence, data integrity is easily maintained in the B_C technique. Similarly, the availability of data to authorized users cannot be denied by others. The blocks are stored in a decentralized network. So, the members involved in the networks are able to access the information without any delay. Thus, the data availability, integrity, and confidentiality are achieved in the proposed system.

7. Conclusion

The Decentralized Medical Network and Patient Record Management platform are built using the smart contracts on the decentralized blockchain. Patient Record Management gives patients control over their medical records in an authoritative manner. The Doctor Network allows doctors to form their own medical network in a secure way and allows them to share their diagnosis and inferences from medical report in a more decentralized way with details stored in smart contracts. These inferences may help remote doctors with critical surgeries to be performed on patients. Files shared or uploaded on both platforms are done through IPFS.

IPFS makes the storage of all medical reports secure and the storage decentralized by distributing the files in packets throughout the network. Thus, this platform serves as a perfect system for all medical record and network-oriented work. It can be adopted for all types of blockchain networks. Though the AES encryption scheme is used for encryption of the IPFS hash, the patients' records can be made more secure using oracles. The Huffman code-based data compression technique reduces block size before storing the PMR into block, reduces storage space requirement, increases scalability, and avoids block size storage issues. The patients will also be provided with the facility to upload multiple report files as a single record, since a single patient could be associated with multiple diseases, and the same IPFS hash can be used to share multiple files under a single directory. In future, the data retrieval and related issues are going to be analyzed in different aspects.

References

1. Sumathi, M., S. Sangeetha. Blockchain Based Sensitive Attribute Storage and Access Monitoring in Banking System. – International Journal of Cloud Applications and Computing, Vol. **10**, June 2020, Issue 2, pp. 77-92.
2. Sumathi, M., N. Vijaya Raj, S. P. Raja, M. Venkatachalapathy. Blockchain Based Adaptive Resource Allocation in Cloud Computing. – Brazilian Archives of Biology and Technology, Vol. **65**, 2022, pp. 1-19.
3. Sumathi, M., N. Vijaya Raj, S. P. Raja, M. Rajkamal. Internet of Things Based Confidential Healthcare Data Storage, Access Control and Monitoring Using Blockchain Technique. – Computing and Informatics, 2022, pp. 1-30.

4. Sumathi, M., M. Rajkamal, B. Gomathy, I. Infant Raj, Sushma, D. Swathi. Secure Blockchain Based Data Storage and Integrity Auditing in Cloud. – Turkish Journal of Computer and Mathematics Education, Vol. **12**, 2021, pp. 159-165.
5. Madine, M. M. Blockchain for Giving Patients Control Over Their Medical Records. – In: IEEE Access. Vol. **8**. 2020, pp. 193102-193115. DOI: 10.1109/ACCESS.2020.3032553.
6. Akkaoui, R., X. Hei, W. Cheng. Edge Medi-Chain: A Hybrid Edge Blockchain-Based Framework for Health Data Exchange. – In: IEEE Access. Vol. **8**. 2020, pp. 113467-113486. DOI: 10.1109/ACCESS.2020.3003575.
7. Omar, A. A. A Transparent and Privacy-Preserving Healthcare Platform with Novel Smart Contract for Smart Cities. – In: IEEE Access. Vol. **9**. 2021, pp. 90738-90749. DOI: 10.1109/ACCESS.2021.3089601.
8. Zhuang, Y., L. R. Sheets, Y.-W. Chen, Z.-Y. Shae, J. J. P. Tsai, C.-R. Shyu. A Patient-Centric Health Information Exchange Framework Using Blockchain Technology. – IEEE Journal of Biomedical and Health Informatics, Vol. **24**, August 2020, No 8, pp. 2169-2176. DOI: 10.1109/JBHI.2020.2993072.
9. Ilhaam, O., R. Jayaraman, M. Debe. Automating Procurement Contracts in the Healthcare Supply Chain Using Blockchain Smart Contracts. – IEEE Access, 2021, p. 99.
10. Zhang, P., D. C. Schmidt, J. White, G. Lenz. Chapter One – Blockchain Technology Use Cases in Healthcare. – In: P. Raj, G. C. Deka, Eds. Advances in Computers. Vol. **111**. Elsevier, 2018, pp. 1-41.
11. Griggo, K. N., O. Ossipova, C. P. Kohlios, A. N. Baccarini, E. A. Howson, T. Hayajneh. Healthcare Blockchain System Using Smart Contracts for Secure Automated Remote Patient Monitoring. – Journal of Medical Systems, Vol. **42**, 2018, No 7.
12. Sharma, R., C. Zhang, S. C. Wingreen, N. Kshetri, A. Zahid. Design of Blockchain Based Precision Healthcare Using Soft System Methodology. – Industrial Management and Data System, Vol. **12**0, 2019, pp. 608-632.
13. Pongnumkul, S., C. Siripanpornchana, S. Thajchayapong. Performance Analysis of Private Blockchain Platforms in Varying Workloads. – In: Proc. of 26th International Conference on Computer Communication and Networks (ICCCN'17), Vancouver, BC, Canada, 31 July-3 August 2017, pp. 1-6.
14. Nihar Ranjan Pradhan, Akhilendra Pratap Singh, SahilVerma, Kavitha, Navneet. A Novel Blockchain Based Healthcare System Design and Performance Benchmarking on a Multi-Hosted Testbed. – Sensors, Vol. **22**, 2022, 3449.
15. Sharma, V., A. Gupta, NajamUlHasan, M. Shabaz. Blockchain in Secure Healthcare Systems: State of the Art, Limitations, and Future Directions. – Security and Communication Networks, Vol. **22**, 2022, pp. 1-15.
16. Pariselvam, S., M. Swarnamukhi. Encrypted Cloud Based Personal Health Record Management Using Des. Scheme. – In: Proc. of IEEE International Conference System, Computation, Automation, Network, (ICSCAN'19), March 2019, pp. 1-6.
17. Wang, C. J., X. L. Xu, D. Y. Shi, W. L. Lin. An Efficient Cloud-Based Personal Health Records System Using Attribute-Based Encryption and Anonymous Multi-Receiver Identify-Based Encryption. – In: Proc. of 9th International Conference P2P, Parallel, Grid, Cloud Internet Computing, November 2014, pp. 74-81.
18. Abid Haleem, Mohd Javaid, Ravi Pratap Singh, Rajiv Suman, Shanay Rab. Blockchain Technology Applications in Healthcare: An Overview. – International Journal of Intelligent Networks, 2021, pp. 130-139.
19. Salah, K., M. H. U. Rehman, N. Nizamuddin, A. Al-Fuqaha. Blockchain for AI: Review and Open Research Challenges. – IEEE Access, Vol. **7**, 2019, pp. 10127-10149.
20. IbrarYaqoob, K., Raja Jayaraman, Yousof Al-Hammadi. Blockchain for Healthcare Data Management: Opportunities, Challenges, and Future Recommendations. – Neural Computing and Applications, 2022, pp. 11475-11490.
21. Ma, M., G. Shi, F. Li. Privacy-Oriented Blockchain-Based Distributed Key Management Architecture for Hierarchical Access Control in the IoT Scenario. – IEEE Access, 2019, pp. 1-15.

Received: 28.08.2022; Second Version: 12.10.2022; Accepted: 20.10.2022