A Scalable Block Chain Framework for User Identity Management in a Decentralized Network

R. Geetha · T. Padmavathy · G. Umarani Srikanth

Accepted: 20 October 2021 / Published online: 30 October 2021
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract
In a decentralized network every user makes use of personal identity details at different places for various services and these details are shared with third-parties without their consent and stored at an unknown location. Organizations like government, banks and social platforms are considered to be the weakest area in the current identity management system as they are vulnerable which in turn leads to compromising billions of user identity data. Block chain based User Identity Management is a solution which provides a decentralized environment that manages the user identity data and their related Know-Your-Customer documents in a distributed ledger. All the transactions of the network are stored in the block, which is a type of a data structure and these blocks are validated using the powerful consensus algorithms and linked to form a block chain. Smart contracts will act as an interface between the client and the block chain network. User’s information cannot be provided to any third party vendors without the explicit consent of the user. This paper proposes a framework for User Identity Management using Block chain technology in a decentralized network. The proposed framework ensures a high level privacy and security for the personal identity details and the documents. In addition to that the performance analysis of the framework is presented in terms of Transaction immutability, Mining Resource, Memory utilization and Difficulty Variation. The results of the analysis proves that the proposed framework performs better in terms of all these factors.

Keywords Block chain · Authentication · Security · Privacy · Peer to peer (P2P) network

1 Introduction
Today in this digitalization era users can able to access any type of digital services from anywhere. This ubiquity has also caused numerous challenges to the clients as well as the service providers in security aspects. Each and every year there is an increase in the cyber related cases or crimes due to the insecure environment while providing or handling these
types of digital services. Among all of them one of the most important issues and the start-
ing point of these types of things is the lack of security to handle the identity data of the
individuals [1]. This form of inefficiency will end up with some unpredictable effects to not
only to the single individual and also for the other persons related to them.

In order to provide any type of digital services to the users in either online or offline
mode some identity may require to identify them. This identity may differ for different type
of services based on the requirement. Personal identity details of the individual and their
related KYC documents are mostly used by majority of the service providers to provide
services and large number of organizations to utilize the identity data for some purpose [2].
Large amount of the services provided by the service providers is not possible without this
type of digital identity. Some of organizations like Government, Banks, Insurance agency,
Hospitals, University and Schools, private organization etc., use identity data and docu-
ments for variety of services to their citizens, customers, patients and students respectively
[3]. It is the responsibility of the corresponding providers to handle this identity data and
documents in a secure manner. There exists a wide range of security mechanisms or algo-
rithms in various domains to protect the information [4–6].

Majority of the current identity management solutions are centralized systems, which is
a client—server architecture. Billions of digital identity and related documents are stored
in a single point (server) [7]. Since all data are stored in a single location, there is a lot of
possibility for security compromise using the variety of security attacks which leads dan-
ger to the individual person, since the individual may use the same identity to utilize dif-
ferent type of services. Similarly, at any point of time if the server which is handling these
data is crashed then it may lead to loss of the valuable data [8].

Many of the organization use some third-party service providers to handle the personal
identity data and KYC documents. Majority of third-party service providers are not so
strong in base to handle such valuable data and most of them are found to be a centralized
one [9]. Another issue with these third parties is the reliability or trust on them for our
data. There are lot of possibilities that these entities can share our personal identity docu-
ments with other unauthorized organizations and further that organization may share that
data with other persons [10]. Therefore, without the permission of the owner of data, docu-
ment data may be flooded to many external persons.

Lack of strong validation and verification of the identity data in identity management
may lead to inclusion of lot of fake identity details and also duplication of the data. Both
this redundant and faulty information may reduce the efficiency of services provided to
the clients. Moreover, identity data are stored directly in centralized location in a human
readable form. So, an attacker can able to know about the particular identity data or docu-
ment without severe efforts once the attacker locates the data. Even though some identity
schemes provide encryption and hash-based security for identity data, most of the algo-
rithms used are easily compromised using common attacks [11].

Block chain is a distributed, decentralized ledger-based technology which stores the data
in a data structure called blocks. It provides a secure way of storing, managing, accessing
the data in the blocks. A block may consist of number of fields which makes the block as
secure one. All the blocks are serially linked together to form a chain, and these blocks
uses the previous block hash field to form a strong chain. Each block has a field called
Timestamp which denotes the time in which the particular block is added to a chain [12].
It is based on peer-to-peer network which is a distributed network in which all the par-
ticipants are connected together without any central entity. Communication between these
participants of the network is done by message passing mechanisms. Every participant of
the network is known as nodes and there are some special dedicated nodes in this network.
called miner nodes. The role of these miner nodes is to validate the blocks which are added to the block chain using necessary algorithms.

Each and every node of the block chain P2P network will have a copy of the block chain ledger as shown in Fig. 1. This ledger consists of the updated block chain such that any transactions performed on a block chain network will be recorded in each copy of the block chain.

The three main characteristics [13] which make the block chain as a unique strategy to handle the current digital technologies are (a) Decentralization where there is no centralized entity or any third-party sources to store the data or performing any network control activities. Due to the decentralized environment of block chain each node of the network will have an updated block chain ledger so any crash or failure in one of the working nodes will not lead to any trouble since data can be recovered from the ledger of other nodes. (b) Immutability where data that are stored using block chain technology are immutable, that it is very difficult to alter the data present in the chain. Each and every block in the block chain is linked using previous block hash and it also has a special numeric field called Nonce. If an attacker wants to modify any data in a block it is mandatory to find this nonce value of the particular block. Then if the attacker tries to update any incorrect data it may lead to change in the block hash, so all the subsequent blocks which are linked using the hash also changed. This leads the attacker to crack each block nonce value and change the Hash which is practically difficult. An attacker may need to spend lot of resources or money based on the consensus algorithm used. (c) Transparency where all the participants of the network can able to record the transactions performed in a network in their ledgers. Due to the strong agreement between the nodes of the network, some of the main issues of current identity management such as duplication of data and the fake details can be eliminated. A miner node can able to reject a block which is found to be faulty or invalid one using this transparency between the nodes.

Consensus algorithms ensure that only a valid transaction or block is added to a block chain. These algorithms use some strategy for finding a specific value called Nonce. This process is known as Mining and it is performed by special nodes called miner nodes. After successful mining the miner will be awarded with the native coins. The main objective of the proposed algorithm [14] is to solve a given function or a problem by using the available block hash value and find the corresponding Nonce. Miner nodes will mine the block using the available resources and add it to the block chain, so some resources such as power, storage is required in order to find this nonce value. The difficulty target will be changed.

Fig. 1 Block chain P2P network
for every 2016th block of a chain, so the algorithm becomes more complex to mine. This algorithm also tries to find the nonce value of the block but instead of using plenty of resources to mine a block it uses a different strategy. Each and every miner must hold some native platform coins to mine, so only miners with optimal coins for mining will allow to mine the block. But it is practically expensive for a user to hold that much of coins to mine a block [15].

1.1 Significance of the proposed work

Since the users have the access to digital services anywhere at any time in this digital world there is a need to protect these information from suspicious users. Moreover cyber threats are increasing day by day due the insecure environment during the access to the digital services. It is observed that in the recent days there is a lack of security in handling the identity data of the individuals. Since there is a gap in addressing these security issues and moreover block chain technology is gaining its popularity in recent days, we can address the gap with the usage of this technology for protecting the individual’s identities in digital mode.

The major objectives/contributions of this paper includes.
To deliver a framework for User Identity Management in a decentralized network.
To store, manage and access the personal identity details and their documents.
To provide an Ethereum platform for signing the transactions performed in a block chain.
To address the privacy and security issues for the personal identity details.
To offer superior performance in terms of Transaction immutability and Mining Resource.
To provide an enhanced Memory utilization and Difficulty Variation.

The report is organized in such a way that chapter 2 provides the related work. Chapter 3 briefly discuss about the proposed Framework for User Identity Management. Performance analysis of the proposed framework is presented in chapter 4. Finally, Chapter 5 concludes the presented framework.

2 Related Work

In this paper [16] the authors have discussed about the Identity management which allows sharing the user identity information to different entities based on the requirement among several entities and across different domains. Nowadays, Majority of clients prefer to reuse a single digital identity instead of changing constantly their credentials for different digital services. Block chain based user identity ensures a high level transparency to all the participants of the network, so that sharing of data between the participants of the network is transparent to all the participants and it also ensures a secure way of using single identity.

In this work [17] the author explained that users need to depend more on the intermediate third party identity provider or any service provider to handle the identity details and their related documents. Most of the time, identity information is not updated or deleted after it has become inapplicable and these things may lead to entry of faulty or redundant identity information. They stated that block chain solution eliminates the
need of the third party to provide the identity services instead of that all the nodes like bank, government are known to each other in a distributed manner. Since each node has a distributed ledger, it is possible to monitor the transactions in all possible nodes and allows only valid data to be stored.

The authors [18] portrayed the issues with the current digital identity management in which if an organization needs a certain proof for background verification, then the client may provide a complete document which may contain some additional personal details rather than main requirement. The solution given by them ensures that only requested atomic data is provided to the organization. It is also possible for the owner of the identity data to reject the unnecessary additional data request by the organization. This provides a secure way of filtering the irrelevant details.

Here the author [19] described about OAuth 2.0 protocol that deals with the basic security issues by introducing an additional layer called authorization layer which separates the role of the client from identity owner. Instead of using the direct credentials of owner it uses a secret token to access the details. Even though the credentials of the user are secure, the details as well as token are still stored in a centralized location. They proved that decentralization can securely handle the identity credentials and the related documents in a distributed manner and can also be recovered easily.

The author [20] described about how different government organizations store their relevant details at their own limited databases and they depend on the other organizations for some required data which is not available with them. He described the usage of block chain based identity in storing the identity details in a ledger as blocks and these ledgers are distributed to all the participants of the network. He concluded that each and every node will have a same copy of the identity data and it can be used by organizations with permission.

The authors in this work [21] suggested building a powerful digital identity system which allow us to handle the identity details for lifetime without any centralized entity or third party entity and it ensured that the data should be shared only with permission. They have implemented the above requirements using block chain solution which ensures that all the identity data and documents are stored in a secure ledger and data is shared only with explicit consent of owner.

Proof of Work algorithm described by the authors [22] utilizes the resources of system for mining process and Proof of Stake algorithm uses the platform based coins for mining. But both this algorithm has a common attack called 51% attack and these attacks are practically difficult. They have concluded that even though the Proof of Work algorithm suffers from huge resource utilization for mining process, it is not expensive as Proof of Stake since it depends upon the coins. Proof of Work is also a secure way of mining compare to Proof of Stake algorithm.

The authors [23] explained that the transaction speed of a blockchain technology can be varied in any range either low or high based on the usage of the block chain type and their related configurations. If the speed of the transaction is found to be low, it mandates the evaluation of the consensus algorithm. This block chain based identity solution uses the minimum nonce value with a complex difficult target value. This makes the block chains difficult to mine as well as to control the time taken for mine a block in the block chain.

This paper [24] explains about the assets that are stored in permission-less public block chain model in which the unknown nodes can easily join and leave the network. It is difficult to deal with these temporary nodes as it leads to lack of trust on nodes. Hence the authors proposed a block chain based identity which is a private permissioned blockchain which can able to handle these nodes more effectively and securely in a controlled manner.
They further proved that the proposed model ensures valid transactions are performed by valid authenticated nodes.

In this paper [25] the authors discussed about the privacy in the government organizations. Due to the significant features of block chain, it can be used by the government organizations to provide all type of digital services to the citizens and ensures a high level privacy to identity data. The solution provided by them can also be used by government as well as other giant private organizations to handle the employee related background details.

3 Proposed Block Chain Framework for User Identity Management

The proposed system is a block chain based solution for storing, managing and accessing the personal identity details and their related KYC documents. Using a web based dashboard, users can add their relevant identity details and upload the KYC documents. Instead of storing the user identity data in the conventional databases, this solution allows users to store all the identity information in a block (a data structure with certain number of fields). These blocks are signed by using the cryptographic algorithms. After signing, these blocks are forwarded to a block chain network where the mining operation take place and these blocks will be linked together to form a chain structure called as block chain.

The peer to peer network consists of number of nodes which are permitted to use the identity information. Any nodes can able to join the network or leave the network. Some of the ordinary nodes of this solution are Government, Bank and University. Each and every node will own its public and private key pairs generated. These keys are used to sign the transaction and send it to the network for mining. Among these nodes there may be one or more mining nodes which have a special role to validate the blocks. Each and every node will have a ledger and this ledger consists of the updated block chain. These ledgers will record all the activities related to the block chain network. Users and other organizations can able to monitor the entire block chain history such as block properties, transaction details, mining details, organization account details and the block content in hashed form. It also shows the details of the similar smart contract deployment related history in the block chain.

Any organization which is interested in an identity data or a document will send a transaction request to the block chain. This request will be validated by miner nodes of the network and also with an explicit consent approval given by the owner for the legitimate transaction. After the approval of the user, the particular identity data requested will be returned to the network as transaction. This data will also undergo a mining process and provided to the organization requested. A user has the right to reject a request transaction if he is not willing to share the data to a particular organization.

This solution is a private permissioned block chain application, so each and every node need to provide their own identity to join the network. This feature makes block chain as a secure way of handling the data within the known organizations. Due to the usage of the distributed ledger the activities of the user node or other organization node is transparent to all which ensures that only a valid user performs a valid transaction.

In a block chain based solution in order to perform any transaction it may require some gas. A gas is nothing but a unit required to perform a transaction. This gas can be obtained by using the native money of the Ethereum called as Ether (ETH). Each account (organization) will have a certain amount of ether with them and to perform a transaction these organizations must use some ether from their account.
Since we use a private block chain for this personal identity management, these Ethers are just virtual money within the network. But the real world Ether is very expensive and it may need some real money to buy the Ether for an account.

The algorithm proposed to implement the block chain based identity solution is as follows;

Step 1 Create a new package for carrying dependency of block chain.
Step 2 Create a new config file to serve the client pages to browser.
Step 3 Create a new smart contract for handling the identity data and documents.
Step 4 Migrate the required contracts to the block chain.
Step 5 Create a new file and initialize a get blockchain environment.
Step 6 Create new account.
Step 7 Start the block chain console and start the mining operation and generate new blocks.
Step 8 Create a new client page.
Step 9 Create a file which is used to obtain the block chain details in client browser.
Step 10 Launch the application.
Step 11 Now the application is launched and synchronized.

The framework shown in Fig. 2 explains the way the identity data or KYC documents which are handled in a block chain based identity management. It consists of different components for performing different actions or services within the block chain.

Users will provide their personal identity details and their KYC documents through the client interface (web application). Organizations can also request the data or document through another interface.

Any transaction performed in a block chain is signed by using these pair of keys. Ethereum platform uses an Elliptic Curve Digital Signature Scheme (ECDSC) for signing the transactions performed in a block chain. The process in this algorithm includes the generation of two pair of keys using the available domain parameters and a random number. A random number generation is started and a numeric scalar value is obtained which is the private Key. Using private key and the other domain values such as G(x, y) public key is generated. Then two signature components ‘r’ and ‘s’ are calculated using the following formula represented as;

\[
\begin{align*}
Q(x, y) &= d \times G(x, y) \\
(x_1, y_1) &= k \times G(x, y) \mod p \\
r &= x_1 \mod n \\
s &= (k^{-1}(h(m) + d \times r)) \mod n
\end{align*}
\]

Hashing converts a given identity data or a document address to a fixed length hash based on the type of hash function. Hashing scheme is the one of the core security features of the block chain based solution. The way this hash is used in block chain is very complex compared to the conventional applications.

The public key encryption algorithm used by the Ethereum platform is found to be “KECCAK-256 a sponge based hash function”. There are 7 available widths for permutations denoted by the given closed set B and its width values are;
The keccak \([r,c]\) is a sponge function used for hashing the given transaction input where ‘r’ and ‘c’ are rate and capacity respectively. The function is organized as

\[ n = 12 + 2 \log_2 l \]

The number of rounds ‘n’ required for applying the sponge function is represented as \( n = 12 + 2 \log_2 l \) and the corresponding sponge function for padding is defined as the below definition;

\[ B \in \{25, 50, 100, 200, 400, 800, 1600\} \]

The 5 \times 5\) lane matrix of length \( w \Rightarrow (1, 2, 4, 8, 16, 32) \Rightarrow w = 2^l \)

Fig. 2 Block chain framework for user identity management
In the above function the message $M$ is represented as $M_{bytes}$ and another value of function $M_{bits}$ is 7 trailing zeros. Finally, the variable ‘p’ returns the padding array calculated for the given input message. Any actions performed on a block chain network are called a Transaction. All these transactions are stored in the Blocks. All the transactions initiated must be signed before forward it to the network. Thus, it is difficult to alter the data which is present in the hash due to the complex strategy used for hashing.

A block is a data structure which is used to store the relevant identity and document related data. A block has different fields associated with it for making it as a secure way of storing data. It has two parts: Block header and block body. All the signed transactions signed are stored in these blocks. Now these blocks will be submitted for mining operation. An Ethereum wallet is used to transfer the Ethers from one account to another account. A wallet stores the pair of keys of a particular participant of the network. Using the public key addresses ethers will be transferred and private key addresses are used to receive the ethers rewarded for successful mining operation.

Consensus ensures that all the nodes of the network agree to allow the particular block to be part of a block chain. Once all nodes agree, a block can be added to a block chain ledger. If any block which is found to be an invalid block can be rejected by the participants. This solution uses the Proof of Work algorithm for mining operation. If a user uploads a data, it will be formed as transaction and this transaction will be stored in the block body. Now these blocks should be mined by the miner nodes of the network. A network may have more than one miner nodes and the miner who mines the block first will be rewarded with ether. This mined information will be shared with the other nodes of the network for a common consensus and after this these blocks will be linked to block chain. As per the proof of work algorithm miner nodes need to identify the nonce value of the particular block fixed. A miner node will be provided with other parameters such as block hash to compute the nonce.

A smart contract is a self-executing code deployed to a block chain to handle the transactions initiated by the individuals or other organizations of the application. It will provide the way the particular data should be handled in a block chain in a secure way. These contracts are found to be an immutable code, so if any change or updating needed in these contracts then they should be redeployed in the block chain. These smart contracts eliminate the need of the intermediate server to handle the data provided by the users.

Distributed ledgers are used to store the mined and validated blocks in the consensus layer. The mined blocks will be added to the longest chain using the previous block hash as linking field. Every participant will have this block chain copy to ensure the transparency. Similar to storing the data and document hash to a block chain, identity data and
related documents can be retrieved from the block chain in a reverse manner. The user required data block is retrieved, mined and provided to the client.

The Identity block structure as shown in Fig. 3 consists of two parts; block header and block body. Block header consists of the block related attributes and block body consists of set of transactions.

Some of the important fields of a block structure and what are their impacts in this block chain application are; Block Version that provides the version in which the block chain is implemented. It provides a set of features and validation rules for the particular block chain. Nonce is a 4 byte random integer value which is started with 0. The complexity of the nonce can be realized based on the difficulty target. A miner needs to find this particular nonce value in order to validate a legitimate block. Difficulty Target is a value which ensures the complexity of the mining operation. If the difficulty is low, then blocks can be easily mined and if the difficulty level is high then it is hard to mine new blocks. This difficulty value will vary for each and every block during mining operation by corresponding formula;

$$\text{block_diff} = \text{parent_diff} + \text{parent_diff} \div 2048 \times (1 \text{ if } \text{block_timestamp} - \text{parent_timestamp} < 13 \text{ else } -1) + \text{int}(2 \times ((\text{block.number} / 100000) - 2)),$$

Timestamp denotes the time in which the particular block is created and it also one of the parameters used to calculate the difficulty parameter of block chain. Merkle root hash makes use of Merkle tree which is a binary search tree which is used to store the identity transactions of the block body. These trees will convert the provided identity data to hashes and store it in a tree structure. All levels of the tree are combined as hashes and this hash is combined with the upper level of the tree to form a new hash. Finally, a single root hash is obtained. This is known as Merkle root hash and this hash is stored in the header of block chain. The way this hash formed makes the complexity of altering the data in a block chain. Previous Block Hash is one in which every block has a block header and the fields

![Structure of identity block](image-url)
associated with this header are combined to form a hash value which is known as a “block hash”. For a new block the “previous block hash” field will have the block hash of the previous block. The first block of the block chain is known as a genesis block. Hence, the previous hash of the Genesis block is 0. If there is a block ‘n’ then the previous hash is the hash of block n-1.

Figure 4 shows the flow of data within a block chain. As per the above diagram the user, organization like banks, universities and bank are represented as entity. It includes enrollment, creation and validation of blocks and the block chain ledger.

4 Performance Evaluation

4.1 Proof of Work Analysis

The proof of work algorithm can be briefly analyzed based on the time taken for mining of the blocks. Complexity of the block mining depends upon the different block parameters such as Nonce and Timestamp. Nonce is a 4 byte integer which should be cracked by the miner nodes to find the particular block hash and the difficulty of this nonce is determined by another parameter called ‘difficulty target’. Based on the difficult target it is very complex to mine a block.

Figure 5 shows performance of mining of blocks. The mining of blocks also strongly depends on the time taken for synchronization of the different nodes of the network and also the available resources of the miner (Since proof of work algorithm depends on the heavy
Even some blocks initialized with a low level difficulty also can take large amount of time to mine in a private Ethereum block chain. As the block size increases difficulty of the subsequent blocks will also be further complicated (upto 2016th block) which makes this block chain as a secure decentralized environment for storing the data.

### 4.2 Transaction ETH Analysis

As per the block chain principles any transaction or smart contract which should be executed with some gas. Gas is a unit of cost or an effort required to execute a transaction in a block chain which is always represented in Gwei unit. This gas can be paid as equalized ETH which is a native digital currency of Ethereum. Miners who mine the particular block will be pay the ETH and the particular miner will also be rewarded with some ETH for successful mining of a block. The maximum amount of gas which should be spent by an application can be fixed by the user during initialization of a network.

From Fig. 6 it is found that the transaction fee for the transactions vary in different situations based on numerous features such as type of data load (either a contract creation or any simple transaction), network related things etc. A transaction can also be failed if a miner could not pay a high gas to mine a transaction or a lower than the gas limit to mine a transaction. By default, any transaction executed on Ethereum needs a minimum gas amount of Rs. 21000.

### 4.3 Difficulty Variation Analysis

Difficulty parameter of block data structure ensures how securely a data is managed in a block chain. In the previous Mining vs block analysis, it shows that how mining operation depends upon nonce value which is because of the difficulty variation value. Difficulty adjustment formula of Ethereum;
Thus, the difficulty value is obtained by some general calculations which uses the parameters from the current new block as well as the previous block such as timestamp and previous difficulty values. If the value obtained in the if clause is 1 then difficult will increases otherwise if it is − 1 then difficulty decreases and difficulty will be stable one for 0. This variation is illustrated as graphically for 5 blocks mined in geth console and the difficulty is assumed as 170 for 1st block (without constant difficulty).

This adjustment algorithm uses a technique called ‘difficulty bombing’ when the block number of the chain is greater than 100,000 and the algorithm start to increase the difficulty value exponentially which is shown in Fig. 7.

![ETH spent for Transaction](image1)

**Fig. 6** ETH spent for 5 identity data upload transactions

\[
\text{block\_difficulty} = \text{parent\_difficulty} \\
+ \frac{\text{parent\_difficulty}}{2048} \\
+ (1 \text{ if } (\text{cur\_block\_timestamp } - \text{parent\_timestamp}) < 13 \text{ else } -1) \\
+ \text{int}(2^*(\text{block\.number} // 100000) - 2)
\]

Thus, the difficulty value is obtained by some general calculations which uses the parameters from the current new block as well as the previous block such as timestamp and previous difficulty values. If the value obtained in the if clause is 1 then difficult will increases otherwise if it is − 1 then difficulty decreases and difficulty will be stable one for 0. This variation is illustrated as graphically for 5 blocks mined in geth console and the difficulty is assumed as 170 for 1st block (without constant difficulty).

This adjustment algorithm uses a technique called ‘difficulty bombing’ when the block number of the chain is greater than 100,000 and the algorithm start to increase the difficulty value exponentially which is shown in Fig. 7.

![Difficulty Variation](image2)

**Fig. 7** Difficulty variation analysis
4.4 Mining Resource Analysis

Since the proof of work algorithm uses resources heavily to mine a particular block to a canonical chain, it becomes mandatory to manage and analysis the performance of the available computer resources. Ethereum mainly use two strategies for mining operation such as CPU Mining and GPU Mining. Even though the GPU is fast compared to the CPU, due to the expensive nature this project uses the CPU mining to mine the block which uses the available resources such as memory, storage and power. Figure 8 shows the `cmd of geth console` which is running the miner.start (1) to mine the first block of the chain.

Therefore, the disk as well as the RAM has been used effectively by the miner nodes and apart from this some amount of power resource is also spent for the mining as shown in Fig. 9.

4.5 Transaction Immutability Analysis

Every transaction hash of a particular blocks is stored in a Merkle tree in a complex and secure way. This organization ensures a block chain immutability feature and also to restrict the tampering of data with help of other nodes of the network. In this analysis X-Axis is found to be inclusion of blocks with an incremental size of 50 and corresponding Y-axis is just an increasing factor (assumed) to denote the difficulty to tamper the data.
If an attacker tries to tamper transaction in block 47 and the change in the hash of that block will leads to change in all the subsequent block hash values since all are linked with help of previous hash value. So, the attacker needs to change the all subsequent block hash with respect to previous with help of mining operation but it is practically difficult to implement and the other nodes can also revoke the particular transaction performed by attacker if they found some alteration has made on an immutable ledger. As a block chain length increases due to the inclusion of new blocks it will be very difficult for an attacker to change the all subsequent block hashes to avoid the callback of the transaction of other nodes as shown in Fig. 10.

4.6 Security Features

A user who is not interested to share or provide his identity data can be able to reject the particular organization initiated transaction during the meta-mask signing operation. If the user rejects the transaction, then the particular transaction will be terminated. This feature ensures that only a valid transaction take place in a block chain application. Figure 11 shows the Transaction Rejection by Account Holders.

It is a private permissioned block chain which prompts for a password at any time when an account is needed to perform a block chain operation, which is created during
the account creation. So only the authenticated account holders can able to perform a transaction in this block chain. This ensures the privacy and the permissioned nature of application.

This work does not have any third party resource or a server to store the identity details of the individuals. There is no back end language like java or python to handle the data; which was replaced by the smart contract codes which is use to handle the data provided by the client user. So, server is eliminated and decentralization is achieved using the smart contracts. All the nodes of the block chain will have a copy of the ledger and the block chain size of the block chain is same across all the ledgers.

5 Conclusion and Future Enhancement

This block chain based user identity management overcomes the issues with the traditional identity management models by organizing the user details and their documents and storing it in an Ethereum block chain in a decentralized distributed environment. The proposed framework ensures transparency to the owner of the identity data by ensuring that only a valid user/entity of the network can use or share their data within the block chain environment. It stores the data and documents in a decentralized environment with help of smart contracts which avoids the involvement of the third party entities. The performance analysis shows that user data has better privacy and is organized in a highly secure way such that data that is included in a block chain are highly immutable. In addition to that the performance analysis of the framework in terms of Transaction immutability, Mining Resource, Memory utilization and Difficulty Variation has been presented.

This work concentrates more on the secure way of organizing the individual identity details and their KYC documents. In future the proposed work can be extended to deal with the individual biometric details like fingerprint, iris and facial recognition. The minimum file size to be upload in the block chain based identity will be increased with less amount of gas usage. Proof of work consensus algorithm used to implement this block chain solution can be further be improved or switched to new algorithms for secure and optimal performance.

Funding Not Applicable.

Declarations

Conflict of interest The authors that they have declare that no conflict of interest.

References

1. Liu, Y., He, D., Obaidat, M. S., Kumar, N., Khan, M. K., & Choo, K.-K.R. (2020). Blockchain-based identity management systems: A review. Journal of Network and Computer Applications, 166, 102731. https://doi.org/10.1016/j.jnca.2020.102731
2. Kassem, J. A., Sayeed, S., Gisbert, H. M., Pervez, Z., & Dahal, K. (2019). DNS-IdM: A blockchain identity management system to secure personal data sharing in a network. Applied Science, 9, 2953. https://doi.org/10.3390/app9152953
3. Gordon, W. J., & Catalini, C. (2018). Blockchain technology for healthcare: Facilitating the transition to patient driven interoperability. *Computational and Structural Biotechnology Journal, 16*, 224–230. https://doi.org/10.1016/j.csbj.2018.06.003

4. Geetha, R., Madhusudhanan, V., Padmavathy, T., & Lalithasree, A. (2019). A light weight secure communication scheme for wireless sensor networks. *Wireless Personal Communications, 108*, 1957–1976.

5. Geetha, R., Suntheya, A. K., & Srikanth, G. U. (2020). Cloud integrated IoT enabled sensor network security: Research issues and solutions. *Wireless Personal Communications, 113*, 747–771. https://doi.org/10.1007/s11277-020-07251-z

6. Geetha, R., & Thilagam, T. (2020). A review on the effectiveness of machine learning and deep learning algorithms for cyber security. *Archives of Computational Methods in Engineering*. https://doi.org/10.1007/s11831-020-09478-2

7. Chalaemwongwan, N., & Kurutch, W. (2018). A practical national digital ID framework on the Blockchain. *Conference on Electrical Engineering/Electronics, Computer and Telecommunications Technology*. https://doi.org/10.1109/ECTIC.on.2018.8620003

8. SinghRathee, P. T. (2021). A systematic literature mapping on secure identity management using blockchain technology. *Journal of King Saud University - Computer and Information Sciences*. https://doi.org/10.1016/j.jksuci.2021.03.005

9. Steichen, M., Fiz, B., Norvill, R., Shabair, W., & State, R. (2018). Blockchain based – decentralized access control for IPFS. *IEEE Conference on Smart Data, Blockchain, Computer and Information Technology, Congress on Cybermatics*. https://doi.org/10.1109/Cybermatics_2018.2018.00253

10. Nyaletey, E., Parzi, R. M., Zhang, Q., & Choo, K.-K.R. (2019). BlockIPFS – blockchain-enabled inter planetary file system for forensic and trusted data traceability. *IEEE Conference on Blockchain*. https://doi.org/10.1109/Blockchain.2019.00012

11. Sayeed, S., & Gisbert, H.-M. (2019). Assessing blockchain consensus and security mechanism against the 51% attack. *Applied Sciences*. https://doi.org/10.3390/app9091788

12. How to Build an Identity application using the Blockchain. Available: www.deveteam.space/blog/how-to-build-an-identification-app-using-blockchain.

13. Sultan, K., Ruhi, U., & Lakhani, R. (2018). Conceptualizing blockchains: Characteristics and related Applications. In: 11th IADIS International Conference Information Systems.

14. (2020). Go-Ethereum - Building geth without go overflow. Available: www.geth.ethereum.org/docs/install-and-build/installing-geth.

15. Toluhi, A. (2018). Land MarketPlaceDapp. Available: www.medium.com/coinmonks/Ethereum-land-marketplace-dapp.

16. Domingo, A. S., & Enriquez, A. M. (2018). Digital identity: The current state of affairs. BBVA Research.

17. Alpar, G., Hoepman, J. H., & Siljee J. (2011). The identity crisis security, privacy and usability issues in identity management.

18. Digital Identity on the Threshold of a Digital Identity Revolution. (2018). White Paper by World Economic Forum.

19. Kaur, G., & Aggarwal, D. (2013). A survey paper on social sign-on protocol OAuth 2.0. *Journal of Engineering, Computer & Applied Sciences, 2*(6), 93–96.

20. Thompson, L. L. (2007). Critical issues in identity management-challenges for Homeland Security. *JSTOR, 47*, 335–356.

21. Sovrin, T. M. (2018). A protocol and token for self-sovereign identity and decentralized trust (2018). A White Paper from the Sovrin Foundation.

22. Sayeed, S., & Gisbert, H. M., (2018). On the effectiveness of blockchain against cryptocurrency attacks. In: *Conference on Mobile Ubiquitous Computing System and Technologies*, 9781612086767, 9–14.

23. Hughes, A., Park, A., Kietzmann, J., & Brown, C. A. (2019). Beyond bitcoin: What Blockchain and distributed ledger technologies mean for firms. Amsterdam: Elsevier.

24. Zakhari, V., Amiri, M. J., Maiyaa, S., Agarwal, D., & Abaddi A. E. (2019). Towards the global asset management in blockchain systems.

25. Borrows, M., Harwich, E., Heselwood, L. (2017). The future of public service identity: Blockchain. Reform.
Publisher's Note  Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**R. Geetha** received B.E degree in Computer Science and Engineering from Madras University in 1999, M.E in Computer Science and Engineering from Anna University in 2006 and Ph.D., in 2017 from School of Computing and Information Technology, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology (Vel Tech Dr. RR & Dr. SR Technical University), India. She has over 20 years of teaching experience and working currently in S.A. Engineering College as Professor. Her research interests include Wireless Networks, Security Schemes in Wireless networks. She is a Life member of Indian Society for Technical Education (ISTE) and member of Computer Society of India (CSI). She is the author/coauthor of several research papers in international conferences and journals.

**T. Padmavathy** received her B.E Degree in Computer Science And Engineering from Anna University in 2010 and M.E in Computer Science And Engineering from Anna University in 2015. Her research interests include computer and network security, wireless network. She is the author/co-author of several paper in international conference and journals.

**G. Umarani Srikanth** received B.E degree in Electronics and communication from Madras University, M.E., degree in Computer Science and Engineering from Bharathidasan University in 1996 and Ph.D., in Computer Science from Anna University in 2013. She has over 28 years of teaching experience and working currently in Bharath Institute of Higher Education and Research, India as Professor. Her research interests include Soft Computing, Internet of Things, Sensor Networks. She is a Life member of Indian Society for Technical Education (ISTE) and member of Computer Society of India (CSI). She is the author/coauthor of several research papers in international conferences and journals.