Effective privacy preserving in cloud computing using position aware Merkle tree model

Shruthi Gangadharaiah[1], Purohit Shrinivasacharya[2]
[1]Department of Computer Science and Engineering, Siddaganga Institute of Technology, Tumakuru, India
[2]Department of Information Science Engineering, Siddaganga Institute of Technology, Tumakuru, India

Article Info

Article history:

Received May 4, 2023
Revised Sep 30, 2023
Accepted Oct 19, 2023

Keywords:

Cloud computing
Computational time
Position-aware Merkle tree
Privacy preserving
Third party auditing

ABSTRACT

In this research manuscript, a new protocol is proposed for predicting the available space in the cloud and verifying the security of stored data. The protocol is utilized for learning the available data, and based on this learning, the available storage space is identified, after which the cloud service providers allow for data storage. The Integrity verification separates the private and the public data, which avoids privacy issues. The integration of the private data is done with the help of cloud service providers with respect to the third-party auditing (TPA). Earlier, public key cryptography and bilinear map technologies have been combined by the researchers, but the computation time and costs were high. To secure the integrity of the data storage, the client should execute several computations. Therefore, this research suggests a reliable and effective method called position-aware Merkle tree (PMT), which is implemented for ensuring data integrity. The proposed system uses a PMT that enables the TPA to perform multiple auditing tasks with high efficiency, less computational cost and computation time. Simulation results clearly shows that the developed PMT method consumed 0.00459 milliseconds of computation time, which is limited when compared to the existing models.

Corresponding Author:

Shruthi Gangadharaiah
Department of Computer Science and Engineering, Siddaganga Institute of Technology
Tumakuru, India
Email: shruthiindbit@gmail.com

1. INTRODUCTION

Cloud computing (CC) processes a huge amount of data which is collected through IoT. The CC uses network devices for monitoring and controlling the physical entity which helps to make the decisions. It is made to embed resource computing and communication among all the physical devices [1]. Cloud computing requires a wide range of services that includes platform as a service (PaaS), desktop as a service (DaaS), software as a service (SaaS), and infrastructure as a service (IaaS) [2]. There is a traditional storage technology that is directly attached to the redundant arrays, and to the produced independent disks for generating the storage network area. Cloud storage has provided the users having large storage, the space and access to data among independent geographical locations [3]. The private data is integrated with the cloud service providers as well as with the third-party verifier whereas the process of verification is performed to avoid security issues [4]. The security of the multimedia data is important with respect to the research direction and information field. Security in the multimedia has mainly 5 aspects namely content confidentiality, multimedia content integrity, controllability, multimedia, and repudiation [5]. The impact of cybercrime on the Internet is far-reaching, and CC is an attractive target for various reasons. Major providers such as google, microsoft, and amazon possess robust infrastructures to stand against cyber-attacks, but not all cloud services pleased such capabilities. Identifying a provider with vulnerabilities that are easy to exploit makes them a highly visible target for cyber-criminals. Clouds lacking adequate security [6] measures become prime targets for malicious activities, given their architecture allows for simultaneous attacks on multiple websites. Without proper security, a single malicious activity could compromise numerous websites. Cloud computingsecurity encompasses several challenges, including multi-tenancy, data loss [7] and leakage, ease of accessibility, identity management, unsafe APIs, inconsistencies in service level agreements, patch management, and internal threats. Enforcing security measures that provide to the diverse needs of all cloud users is challenging, as different users have varying security demands based on their objectives for using cloud services [8]. While previous research in cloud computing has predominantly focused on aspects such as technological architecture, distinguishing features from similar technologies, and security concerns, the paramount criterion that drives adoption remains security. However, it is essential to acknowledge the increasing integration of various intelligent environments like utility computing, smart data centers, pervasive computing, automation, virtualization, and intelligent networks into our daily lives.

The developed provable data possession (PDP) scheme failed to support dynamic auditing for providing data privacy protection [9]–[11]. The another developed public auditing protocol, which utilizes a binary binomial tree (BBT)-like data structure along with the boneh-lynn-shacham signature-based homomorphic verifiable authenticator (BLS-HVA) modelfaced an unexpected problem after introducing a third-party auditor [12]. The other developed model was efficient for novel auditing methods, in providing security to data in the cloud storage based on convergence and to perform symmetric encryption. However, the model issued data that was lost with the cloud where the data loss process occurred in the infrastructure irrespective of the measures provided by the cloud service providers (CSPs) [13]. Shabbir *et al.* [14] utilized data integrity verification based on the short signature algorithm that provided privacy protection. The model supported public auditing by introducing trusted third party (TPA). An advantage of the developed model was that it solved the problem of reducing the communication overhead and computational overhead, the major challenge faced in IoT storage security. The disadvantage however were the security issues as it was vulnerable to attacks. It was insecure and was unable to resist forgery attacks with key disclosure attacks. The developed model was large for storage overhead and communication overhead and did not implement with public verification. Garg *et al.* [15] developed data integrity auditing in a cloud computing model efficiently. The objective of the developed protocol was to reduce the client's complexity to set the phase based on auditing protocol. The remote data integrity problem was addressed by the auditing protocol which overcame the data privacy and integrity issues. However, the developed model failed to perform data dynamic operations and failed to consider the static data issues, faced in security. Ping *at al.* [16] developed secure identity based aggregate signatures (SIBAS) that performed data integrity schemes as per the requirement. It processed the scheme to resort with the trusted execution environment (TEE). The TEE auditor was used to check the data that was outsourced at the local side. The developed model was secure enough to resist the attack from the other node. The developed model solved the problem of delivering data with the unknown cloud services. The developed model integrated the data which had become potentially vulnerable and was based on the assumption of ideal state. However, the security issues were neglected for computer systems which increased the range of attacks. Pitchai *et al.* [17] developed an availability and integrity verification (AIVP) protocol which was available for predicting the space in cloud. The model verified the integrity and stored the data. The developed model solved the integrity issues for data cloud storage in the cloud with the help of bridge gap techniques. The generated challenges are verified by TPA for cloud computing that overcame the protocol issues. The developed model reduced the issues related to security and replayed the attack and forgery. The diffie-hellman cryptography technique was used for identifying the public key cryptography technique for avoiding cloud computing problems. However, the research issues persisted in the cloud computing, when the accessed data files led to interrupted verifiers, remote servers, disk failure, data files' deletion.

Liu *at al.* [18] developed an effective data integrity with auditing scheme to edge compute on the basis of multimedia data enterprises securely. The existing multimedia security scheme failed to deal with the general issues of security without tackling the data integrity problems. The developed model solved the security issues for providing multimedia security that reduced the privacy leakage issues and their occurrence. The model improved the service quality for the enterprise but several outside threats led to insecure computation. It lacked service managers to compute and determine the data that suffered from the problem of security. Depending on the layered modeling of the security mechanisms, Shabbir *et al.* [14] showed how to offer requirement-oriented health information security utilizing modular encryption standard (MES). In rare cases, layered modeling also led to decreased system performance. As a result, including quantum computing into the research schemes increased its effectiveness and improved its suitability for use with mobile and smart products. This method did not take into account the image-oriented data set; it was always explicitly configured for the encoding and decoding of textual data.An effective sequential convex estimation optimization (SCEO) method was created by Anajemba *et al.* [19] to address this issue and enhance the physical layer (PHY) security in a three-node wireless communications system. The outcomes of the tests showed that the SCEO method provided the best efficiency and improved connectivity for the communication. A quick privacy rate optimization approach for a multiple-input, multiple-output, multiple-eavesdropper (MIMOME) environment, which is relevant for security in IoT and 5G systems, was developed by further improving this study. Given that this was the research's ideal area of study, its importance for telecommunications such as the internet of things and the 5G cellular network should not be understated. To improve the security of cloud data, Jayaprakash *et al.* [20] published cloud data encryption and authentication predicated on enhanced merkle hash tree algorithm. The suggested solution used leaf nodes including a hash tag and a non-leaf node through the use of a database of child hash data to encrypt massive amounts of data. Additionally, it offered effective data mapping and made it simpler to spot changes that were made because of appropriate organization. The created approach allowed public audits to offer a safe cloud storage system while protecting privacy. But however, the suggested technique provided lesser effective huge datasets and insecure data exchange as a one-to-one approach. Lightweight blockchain framework for medical record data integrity was demonstrated by Mardiansyah and Sari [21] to lower the computational cost. The blockchain data was shown using the flask micro web server, and an android application was developed using MIT app inventor to read data from IoT devices. Leading-zero was used as a measure of mining difficulty in the light-weight blockchain to ensure data confidentiality and integrity when constructing a block. Overall, it lasted shorter than the current network for low difficulty levels. At the fifth level of complexity it took longer than ethereum to execute transactions however, it still proved to be faster. Furthermore, the established model was lacking in the ability to computationally determine which data was affected by the security issue.

From the overall analysis, it clearly identified that to provide an effective security to cloud data storage systems, every model is developed using various advanced approaches. Some developed models supported public auditing by introducing trusted third party and solved the problem of communication overhead. Additionally, the remote data integrity problem was addressed by the auditing protocol which overcame the data privacy and integrity issues with the help of bridge gap techniques. The advanced cryptography technique was used for identifying the public key cryptography technique for avoiding cloud computing problems. Similarly, some techniques offered an effective data mapping and made it simpler to spot changes that were made because of appropriate organization. Additionally, including advanced techniques like quantum computing into the research schemes increased its effectiveness and improved its suitability for use with mobile and smart products. On the other side, the developed model failed to perform data dynamic operations and failed to consider the static data issues. More number of developed models are suffering in the time complexity issues. Moreover, the suggested techniques provide ineffective performance and insecure data exchange with computation overhead. In some other, developed models improved the service quality for the enterprise but several outside threats led to insecure computation. The previous researches lacked service managers to compute and determine the data that suffered from the problem of security. So, an effective and reliable plan or strategy is essential to protect data integrity in cases involving public auditing. The suggested solution will assures complete data security and conserves the computed resources of cloud users. To overcome the above stated issues, this research proposes an efficient auditing approach named position aware Merkle tree modelto protect cloud data privacy and data integrity. The major contributions of this manuscript are specified as:

- A trusted TPA is developed to support public auditing for the data user and to prevent the failure of incurring additional overhead for data undertaking.
- The process of signature is performed for reducing the computational overhead based on the hash function. The experimental results show that the computational time and signature time are minimized by implementing the proposed position-aware Merkle tree and the obtained results are superior to the existing models.
- On the other hand, the proposed position-aware Merkle (PMT) method overcomes the storage and complexity problems at the client side and evaluates the performance of the method to provide integrity verification.

The structure of the proposed research work is given as follows: section 2 explains the description of proposed method. Section 3 explains the concept of PMT-based data possession verificationand section 4 discusses the results of the present research work. The conclusion and future work of the present research work is given in section 5.

2. PROPOSED METHOD

In cloud computing, clients might unexpectedly collapse or get overwhelmed by the frequency of integrity checks. Therefore, adding public verifiability to the verification protocol makes sense and is predicted to be useful in achieving cloud computing economies of scale. Therefore, the focus of the current work is on how to create a third-party auditing system that is not dependent on data encryption. The proposed plan emphasizes data storage and facilitates public auditing of data integrity in cloud computing. Additionally, given the popularity of cloud computing, TPAs may be assigned an anticipated rise in auditing work from various users. The proposed method's performance is evaluated by using the various parameters which are described in the results section. The proposed method consists of key generation, encryption, and decryption. The key generation includes RSA cryptosystem that has three types of processes, firstly encryption, then key decryption, and the prime key. The process of encryption contains different hash functions that provide secure hash algorithms (SHA-2 and SHA-3) and message digest algorithm 5 (MD5). Multiple–level hash tree that utilises the Merkle hash tree (MHT) algorithm is used efficiently for identifying the data integrity among various servers [22]. The MHT provides a persistent data structure for mapping the data among the arbitrary length as binary data. The PMT has nodes in each of the trees that know about the relative position at the parent nodes. The chameleon authentication tree (CAT) is known as an important authenticated data structure for data verification in 5G networks. The process of decryption includes the conversion of data encryption back to the original form. In general, the process of reverse is used in encryption where the recipient receives a window or prompt that has the password for entering and accessing the data for encryption.

2.1. Key generation

The present research uses the RSA cryptosystem that includes encryption, decryption, and a prime key. The RSA algorithm does not provide security to attacks that include brute force attack as it is dependent on the RSA for larger prime numbers and is difficult to break. Thus, the proposed RSA generates the key securely and also establishes the public and private keys that are known as the IRSAC algorithm. The IRSAC key generation utilizes 2 random numbers and two prime numbers.

2.2. Encryption

The encryption process is as: i) the encryption process has obtained various components as a public key; ii) the messages are represented in the form of plain text that represents a positive integer; iii) the cipher text computed is represented as $c = m^e \bmod n,$ in which m^e is known as the encrypted message; and iv) the cipher text$c$is used for forwarding it to a user.

2.2.1. Position-aware Merkle tree

The major theme of PMT is retrieving the data quickly and evaluating the position of encrypted data. Additionally, the PMT method overcomes cloud data storage and complexity problems by efficiently verifying the integrity of data in a cloud environment. It achieves this through a hierarchical tree structure where each leaf node represents a data block and each non-leaf node is a hash of its child nodes. This structure ensures data integrity and enables quick verification of any data modification or corruption.The position-aware feature allows the merkle tree to efficiently handle dynamic data changes. When data is inserted, modified, or deleted, only the affected branches and nodes need to be recalculated, minimizing computational overhead. This reduces the complexity of verification operations, making it scalable and suitable for large datasets in the cloud. PMT tree consists of nodes and each of the nodes is aware of the parent nodes' relative positions. Thus, the integrity verification phase is used in the proposed approach, wherein the complete retrieval of MT is not required [23]. The PMT node in the MT is tracked based on the position of the parent node relatively. A node A_i has PMT records that update the position in the tree that is expressed as a 3-tuple $A_i(A_i.p, A_i.r, A_i.v)$. From the expression, $A_i.p$ is known as the relative position of the parent nodes of n_i. $A_i.r$ is the product of the leaf nodes and A_i. The nodes are labeled from the position of left to right at every layer. $A_i.p, A_i.r, A_i.v$ are calculated as shown in (1) to (3):

$$A_i.p = \begin{cases} 0 & if\ A_i \in left\ subtree \\ 1 & if\ A_i \in right\ subtree \\ null & if\ A_i\ is\ root \end{cases} \tag{1}$$

$$A_i.r = \begin{cases} A_r^i.r + A_r^i.r & if\ A_i\ is\ non-leaf\ node \\ 1 & f\ A_i\ is\ leaf\ node \end{cases} \tag{2}$$

$$A_i.v = \begin{cases} h(A_i.p||A_l^i.v||A_r^i.v||A_l^i.v||A_i.r) & if\ A_i\ is\ non-Leaf\ node \\ h(A_i.p||x_i||1) & if\ A_i\ is\ lead\ node \end{cases} \tag{3}$$

PMT has allowed the generation of integrity based on the authentication path for performing data verification based on the direct computation from the root to the tree. The model is rooted without the tree structure querying. The proposed research uses each node's awareness based on position, considering the overall structure of the tree as it has not utilized the data semantics. The model uses the position of data items that are present on the tree. Also, the merkle tree PMT does not provide the data order as it is independent of the integrity proofs [24]. Each node knows the positions and items that have been accessed for the positioning scheme. The memberships proofs are obtained as data items which are checked for their presence. This takes the time of $O(1)$ and thus it is supported as the new data is inserted into the tree. Therefore, the complexities in generation, insertion, and the query runtime are generated which are the same as the merkle tree and Figure 1 shows the PMT structure. The position-aware merkle tree [25] is provided as A_1which is a leaf node which is indexed with the file block x_1 and A_1 is known as the left subtree based on its node A_9 from the parental node. Thus, as per the above formula, $A_1.p = 0, A_1.r = 1$ and $A_1.v = h(0||x_1||1)$. Similarly, the solution is obtained for $A_2 = (1,1,h(1||x_2||1)$ and $A_9 = (0,2,h(0||A_1.v||A_2.v||2))$.

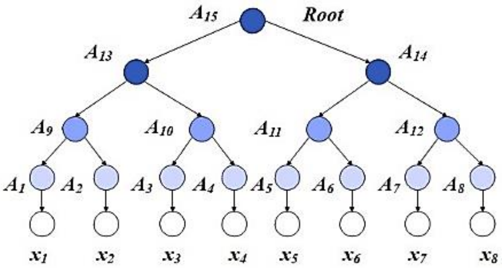

Figure 1. Structure of PMT

3. PMT-BASED DATA POSSESSION VERIFICATION

The present subsection describes the data possession verification which is processed using the PMT. The user keeps the root node $A_{root} = (A_{root}.p, A_{root}, A_{root}.v)$, and the outsource server frames the file block represented as $X = \{x_1, x_2, .., x_n\}\}$. The user needs to check the integrity of the i^{th} file block that is represented as $x_i$that sends the request vector. It is represented as $\{i, "verify"\}$at the server and the server returns the integrity path correspondingly represented as $\{x_i, i\}$. The user can execute the algorithm as $(verify(A_{root}, x_i, i) \rightarrow \{"accept", "reject"\})$ for the calculation of the root node which is represented as A_{root}. This is based on the response $\{x_i, i\}$ that is provided for the server. The user compares the root with the original root which is represented as it is kept locally. In case, $A_{root} = A_{root}$, then the server sends a correct response. Thus, the results are passed to the server for integrity verification and the verification algorithm has been outputted with "$accept$" else it is represented as "$reject$". Therefore, the verification algorithm introduced clarifies the symbols, and the total number of tree nodes are represented as N_t.

3.1. Decryption

The process of converting the encrypted data to its original form is called the process of decryption. Thus, the general process of encoding and reversing is known as encryption and decryption, and message received for the corresponding process in the cloud comes with a prompt or window having a password entered for accessing the data for encryption. The message recipient decrypts the information back to the form of original and readable formats. Further, the messages are passed in a system which are also encrypted. The process is as follows; i) the decryption process utilizes the user's private key (n, d); ii) the message $m = c^d \bmod n$ is computed; and iii) the plain texts extract the message m.

The proposed IRSAC mainly performs key generation, encryption and decryption that are explained as follows:

PSEUDO CODE

```
P and Q are prime numbers and P ≠Q are selected
Calculate n where n =P*Q
Calculate Φ(n)=(P-1) (Q-1)
The Random E is selected in such a way that gcd (Φ(n), E) =1, 1 &lt; E&lt; Φ(n)
The p_publickey and n_publickey are calculated where p. publickey.e =1 (mod Φ(n))
Public key {E,n}
Private key {D,n}
for each p in d
                for each n in d
                If (is Child(p, n))
                   P_material=Hash(p_id,p_publickey)+Hash(n_id,n_publickey)
                End if
       end for
end for
```

4. RESULTS AND DISCUSSION

The proposed model is simulated on a computing system with Intel Core i9 processor, 128 GB random access memory, and windows 10 (64bit) operating system. The proposed method's performance is evaluated by using the parameters of:

- Operation time: is defined as the time interval present among the instants between the occurrence, which specifies the condition of a system and the instant of completion with respect to the specified operation.
- Signature time: the forger acquires multiple message signatures that are performed on the polynomial function from a signed message that has only the public key.
- Computation time: is the length of time required to perform the hash function's computational process.

4.1. Quantitative analysis

Table 1 shows the operation time evaluated in terms of computation time (ms) with respect to the data block size ranging from 0 to 70. The computation time consumed by each of the data blocks varies from 0 to 0.004529 milliseconds. As the size of the data block increases from 0 to 70, the computation time also increases with respect to the data blocks. Figure 2 represents the computation time evaluated by the proposed method.

Table 1. Simulation results by means of operation time/computation time

Data block size	Computation time (milliseconds)
0	0
10	0.004531
20	0.0042915
30	0.004531
40	0.004531
50	0.004291
60	0.00429
70	0.004529

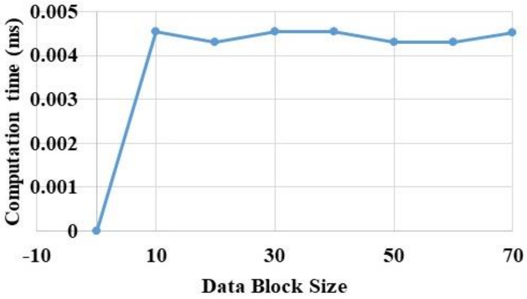

Figure 2. Computation time evaluated by the proposed method

If the size of user's data block is increased, the data blocks numbers are also increases accordingly. The signature time is the time consumed by the signature scheme. The proposed model consumed less signature time than the existing models, which is graphically represented in Figure 3. Table 2 illustrates the simulation outcomes in terms of signature time.

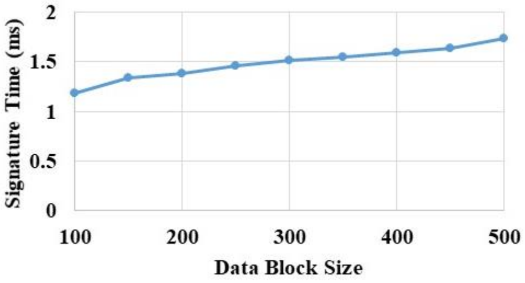

Figure 3. Signature time evaluated by the proposed method

The calculation time varies (increase/decrease) whenever the structure of the tree size changes, for example, if a new branch is extended, time increases, and if a branch is split, it will take less time. The calculation time based on the proposed method is lower and the proposed method uses hash-based operation in the scheme. Table 3 and Figure 4 represent the calculation time of the proposed method, which is evaluated with respect to various data block sizes.

Table 2. Simulation results by means of signature time

Data block size	Signature time (milliseconds)
100	1.18
150	1.34
200	1.38
250	1.46
300	1.51
350	1.55
400	1.59
450	1.63
500	1.74

Table 3. Simulation results by means of calculation time

Data block size	Calculation time (milliseconds)
0	0
10	1.12
20	0.031
30	1.86
40	0.035
50	2.18
60	2.34
70	2.58

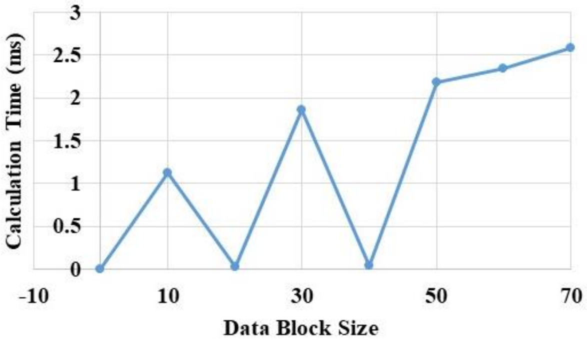

Figure 4. Calculation time with respect the proposed method

4.2. Comparative analysis

Table 4 represents the comparative results between the proposed and the existing models by means of computation time. Ping *et al.* [16] developed a secure identity based aggregate signature model. The developed model overhead in storage with other resources, so it was required to perform a specific task, which resulted with 18 seconds of computation time. There was a lack of service managers in edge computing, which determined that the multimedia data suffered from security threats and consumed 18 seconds of computation time. Pitchai *et al*. [17] developed an effective availability and integrity verification protocol for addressing security and privacy issues in the cloud storage. The simulation results demonstrated that the developed availability and integrity verification protocol consumed 0.30 seconds of computation time, and obtained better performance in terms of other performance measures like throughput and latency. Liu *et al.* [18] implemented an effective data storage system named one-way linked information table for storing the multi-media data enterprise. The developed storage system has obtained higher efficiency of the data recovery, and consumed 18 seconds of computation time. On the other hand, the proposed PMT consumed 0.00459 milliseconds of computation time, which is better compared to the existing models.

Table 4. Comparative results between the proposed and the existing methods

Methods	Computation time (sec)
Secure identity based aggregate signatures [16]	18
Availability and integrity verification protocol [17]	0.3
One-way linked Information table [18]	18
Proposed PMT method	0.00459

4.3. Discussion

In this section, the overall discussion about present research is briefly explained and the results are compared with the previous methods such as Secure identity based aggregate signatures [16], Availability and integrity verification protocol [17] as well as one-way linked information table [18]. Firstly, the RSA cryptosystem that includes encryption, decryption, and a prime key, is used to generate the key securely and also establish the public and private keys. Then the encryption process has obtained various components as a public key. Then, the PMT is used to retrieve the data quickly and evaluate the position of encrypted data. Where, the PMT method overcomes cloud data storage and complexity problems by efficiently verifying the integrity of data in a cloud environment. Then the decryption process begins which is performed using the IRSAC method because the IRSAC method mainly performs key generation, encryption and decryption. Finally, the performance of the proposed PMT method is analysed and compared with existing methods to determine the proposed system's enhanced performance. From the results, it clearly demonstrates that the proposed PMT consumes a minimum computation time of 0.00459 milliseconds for data recovery. Where, the Secure identity based aggregate signatures [16] consumes a computation time of 18 seconds for data recovery and availability and integrity verification protocol [17] consumes a computation time of 0.3 seconds for data recovery and one-way linked information table [18] consumes a computation time of 18 seconds for data recovery.

5. CONCLUSION

The proposed PMT method significantly solves privacy and security issues in the cloud storage system. In the present research work, the proposed PMT method is used for reducing the operation time, signature time, and the computation time of the verification process. The available cloud service providers, and storage space are used for data storage in the multi cloud environment. Therefore, the multi cloud environment increases the number of users and also in parallel, increases the value of throughput. The simulation results demonstrated that the proposed PMT effectively reduced operation time, signature time, and computation time related to the existing models. The proposed PMT consumed 0.00459 milliseconds of computation time, which was better compared to the existing models: secure identity based aggregate signatures, availability and integrity verification protocol, and one-way linked information table.

The PMT rely on hash functions to verify the integrity of the data they store. However, not all hash functions are suitable for use with Merkle trees. This can limit the types of data that can be stored using Merkle trees on a blockchain network. As a future work, the current research can be extended by considering the problem of storing different kinds of data, which can be done by using a novel encryption approach.

REFERENCES

[1] P. Kalia, D. Bansal, and S. Sofat, "Privacy Preservation in Cloud Computing Using Randomized Encoding," *Wireless Personal Communications*, vol. 120, no. 4, pp. 2847–2859, Oct. 2021, doi: 10.1007/s11277-021-08588-9.

[2] A. Razaque, M. B. H. Frej, B. Alotaibi, and M. Alotaibi, "Privacy preservation models for third-party auditor over cloud computing: A survey," *Electronics (Switzerland)*, vol. 10, no. 21, p. 2721, Nov. 2021, doi: 10.3390/electronics10212721.

[3] H. O. Mansour, M. M. Siraj, F. A. Ghaleb, F. Saeed, E. H. Alkhammash, and M. A. Maarof, "Quasi-Identifier Recognition Algorithm for Privacy Preservation of Cloud Data Based on Risk Reidentification," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1–13, Aug. 2021, doi: 10.1155/2021/7154705.

[4] J. Sun, G. Xu, T. Zhang, H. Xiong, H. Li, and R. H. Deng, "Share Your Data Carefree: An Efficient, Scalable and Privacy-Preserving Data Sharing Service in Cloud Computing," *IEEE Transactions on Cloud Computing*, vol. 11, no. 1, pp. 822–838, Jan. 2023, doi: 10.1109/TCC.2021.3117998.

[5] H. Yu, X. Lu, and Z. Pan, "An Authorized Public Auditing Scheme for Dynamic Big Data Storage in Cloud Computing," *IEEE Access*, vol. 8, pp. 151465–151473, 2020, doi: 10.1109/ACCESS.2020.3016760.

[6] I. L. H. Alsammak, M. F. Alomari, I. S. Nasir, and W. H. Itwee, "A model for blockchain-based privacy-preserving for big data users on the internet of thing," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 26, no. 2, pp. 974–988, May 2022, doi: 10.11591/ijeecs.v26.i2.pp974-988.

[7] N. M. Mohammed, L. R. Sultan, A. A. Hamoud, and S. S. Lomte, "Verifiable secure computation of linear fractional programming using certificate validation," *International Journal of Power Electronics and Drive Systems*, vol. 11, no. 1, pp. 284–290, Mar. 2020, doi: 10.11591/ijpeds.v11.i1.pp284-290.

[8] S. Sulthana and B. N. R. M. Reddy, "Machine learning algorithms for privacy preserving in vehicular ad hoc network," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 2, pp. 1021–1028, May 2023, doi: 10.11591/ijeecs.v30.i2.pp1021-1028.

[9] A. K. Leaby, M. Khalefa, M. A. Hasson, and A. A. Yassin, "Secure authentication and privacy-preserving to improve video streaming vehicle ad-hoc network," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 28, no. 1, pp. 480–487, Oct. 2022, doi: 10.11591/ijeecs.v28.i1.pp480-487.

[10] M. A. Al-Shareeda, M. Anbar, M. A. Alazzawi, S. Manickam, and I. H. Hasbullah, "Security schemes based on conditional privacy-preserving vehicular ad hoc networks," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 1, pp. 479–488, Jan. 2021, doi: 10.11591/ijeecs.v21.i1.pp479-488.

[11] K. V. Kumar and J. Harikiran, "Privacy preserving human activity recognition framework using an optimized prediction algorithm," *IAES International Journal of Artificial Intelligence*, vol. 11, no. 1, pp. 254–264, Mar. 2022, doi: 10.11591/ijai.v11.i1.pp254-264.

[12] X. Zhang, X. Wang, D. Gu, J. Xue, and W. Tang, "Conditional Anonymous Certificateless Public Auditing Scheme Supporting Data Dynamics for Cloud Storage Systems," *IEEE Transactions on Network and Service Management*, vol. 19, no. 4, pp. 5333–5347, Dec. 2022, doi: 10.1109/TNSM.2022.3189650.

[13] Z. Yi, L. Wei, H. Yang, X. A. Wang, W. Yuan, and R. Li, "An Improved Secure Public Cloud Auditing Scheme in Edge Computing," *Security and Communication Networks*, vol. 2022, pp. 1–9, Apr. 2022, doi: 10.1155/2022/1557233.

[14] M. Shabbir *et al.*, "Enhancing Security of Health Information Using Modular Encryption Standard in Mobile Cloud Computing," *IEEE Access*, vol. 9, pp. 8820–8834, 2021, doi: 10.1109/ACCESS.2021.3049564.

[15] N. Garg, S. Bawa, and N. Kumar, "An efficient data integrity auditing protocol for cloud computing," *Future Generation Computer Systems*, vol. 109, pp. 306–316, 2020, doi: 10.1016/j.future.2020.03.032.

[16] Y. Ping, Y. Zhan, K. Lu, and B. Wang, "Public data integrity verification scheme for secure cloud storage," *Information (Switzerland)*, vol. 11, no. 9, pp. 376–385, 2020, doi: 10.3390/INFO11090409.

[17] R. Pitchai, S. Babu, P. Supraja, and S. Anjanayya, "Prediction of availability and integrity of cloud data using soft computing technique," *Soft Computing*, vol. 23, no. 18, pp. 8555–8562, 2019, doi: 10.1007/s00500-019-04008-0.

[18] D. Liu, J. Shen, P. Vijayakumar, A. Wang, and T. Zhou, "Efficient data integrity auditing with corrupted data recovery for edge computing in enterprise multimedia security," *Multimedia Tools and Applications*, vol. 79, no. 15–16, pp. 10851–10870, 2020, doi: 10.1007/s11042-019-08558-1.

[19] J. H. Anajemba, Y. Tang, C. Iwendi, A. Ohwoekevwo, G. Srivastava, and O. Jo, "Realizing efficient security and privacy in IoT networks," *Sensors (Switzerland)*, vol. 20, no. 9, p. 2609, May 2020, doi: 10.3390/s20092609.

[20] J. S. Jayaprakash, K. Balasubramanian, R. Sulaiman, M. K. Hasan, B. D. Parameshachari, and C. Iwendi, "Cloud Data Encryption and Authentication Based on Enhanced Merkle Hash Tree Method," *Computers, Materials and Continua*, vol. 72, no. 1, pp. 519–534, 2022, doi: 10.32604/cmc.2022.021269.

[21] V. Mardiansyah and R. F. Sari, "Lightweight Blockchain Framework For Medical Record Data Integrity," *Journal of Applied Science and Engineering (Taiwan)*, vol. 26, no. 1, pp. 91–103, 2022, doi: 10.6180/jase.202301_26(1).0010.

[22] L. Zhou, A. Fu, Y. Mu, H. Wang, S. Yu, and Y. Sun, "Multicopy provable data possession scheme supporting data dynamics for cloud-based Electronic Medical Record system," *Information Sciences*, vol. 545, pp. 254–276, Feb. 2021, doi: 10.1016/j.ins.2020.08.031.

[23] Z. Wang, J. Qin, X. Xiang, and Y. Tan, "A privacy-preserving and traitor tracking content-based image retrieval scheme in cloud computing," *Multimedia Systems*, vol. 27, no. 3, pp. 403–415, Jun. 2021, doi: 10.1007/s00530-020-00734-w.

[24] H. Zhu, Y. Guo, and L. Zhang, "An improved convolution Merkle tree-based blockchain electronic medical record secure storage scheme," *Journal of Information Security and Applications*, vol. 61, p. 102952, Sep. 2021, doi: 10.1016/j.jisa.2021.102952.

[25] Y. Li, M. Tang,"Blockchain-powered distributed data auditing scheme for cloud-edge healthcare system", *Cyber Security and Applications*, vol. 1, p.100017, 2023, doi: 10.1016/j.csa.2023.100017.

BIOGRAPHIES OF AUTHORS

Shruthi Gangadharaiah holds her Bachelor Degree in Information Science and Engineering in 2005 and Masters from SJCE College of Engineering, in Specialization of Network and Internet Engineering, Currently Pursuing his Ph.D. in Visvesvaraya Technological University, Belagavi in Computer Science and Engineering and his research are includes providing security in cloud computing. She can be contacted at email: shruthiindbit@gmail.com.

Purohit Shrinivasacharya got his Ph.D. in the area of Image Processing in the year 2016, He has published research paper to his credits including many conference papers. His research interests include, image processing, computer architectures and cloud computing, presently he is working as associate professor in capacity of Department of Information Science and Engineering in Siddaganga Institute of Technology, Tumakuru. He can be contacted at email: purohithsn@gmail.com.