A NEW MODEL PROPOSED FOR MERKLE TREE BASED KEY MANAGEMENT IN CLOUD COMPUTING

Dharavath Nagesh[1], Dr.Harsh Lohiya [2], Dr. Laxmaiah Mettu[3]

[1]*Research Scholar, Dept. of Computer Science and Engineering*

Sri Satya Sai University of Technology and Medical Sciences,

Sehore Bhopal-Indore Road, Madhya Pradesh, India.

[2]*Research Guide, Dept. of Computer Science and Engineering*

Sri Satya Sai University of Technology and Medical Sciences,

Sehore Bhopal-Indore Road, Madhya Pradesh, India.

[3]*Research Co-Guide, HOD. Dept. of Computer Science and Engineering*

CMR Engineering College, Kandlakoya (V), Medchal, Hyderabad

ABSTRACT

Cloud computing is a concept that refers to an entire enterprise that provides expert preparation power, lightning-fast calculation speeds, and a large display of stage space. The cloud computing movement moved immense volumes of information off of data, which is held by the various cloud providers, into the hands of cloud users, making it possible for them to privatize the information even though they have total control. It's extremely interesting to see how many new vulnerabilities have surfaced during its implementation. This paper offers an openly assured cloud trustable solution which relies on a position-aware Merkle tree. We are provided with a 3-tuple, which tells us about the relationship between the various hubs and isomers, so we don't have to rebuild the whole Merkle tree to obtain the root outcome.

Keywords: Merkle tree, Cloud computing, security, Key management

INTRODUCTION

Cloud computing is the on-demand availability of PC framework properties, most notably data storage (cloud storage) and computing power, without the client performing direct dynamic management. Typically, the term refers to server farms that are accessible via the Internet to a large number of clients. Today's massive clouds frequently have capabilities that are dispersed through many areas by focal staff. If the relationship with the client is generally favorable, it is possible that an edge worker would be assigned [1]. Clouds may be exclusive to a single association (Endeavour clouds) or may be open to several associations (multiple association clouds) (public cloud). Cloud computing is based on asset sharing to achieve scalability and reliability. Cloud computing proponents point out that public and hybrid clouds allow organizations to avoid or significantly reduce upfront IT foundation costs [2]. Additionally, proponents assert that cloud computing enables businesses to get their applications up and running faster, with greater predictability and less assistance, and that it enables IT organizations to more rapidly adjust resources to meet fluctuating and erratic demand, providing burst computing capacity: high computing power at particular moments of peak interest [3].

The concept of Merkle tree was originated by Ralph Merkle in 1979. When simplified, the multi-hub is a tree in which each knowledge branch is called using the branch's hash, and each non-hub is given a cryptographic name. Leaves are the hubs at the base of the network. While using this method, the merkites die when they become over identifying; there's only one set of hubs when all have been formed [4]. The Merkle tree is often called the "root" hash. Each one of the merkle trees is tailored to a specific application. In cryptography and software engineering, a hash tree is defined as a tree where each node is designated by a hash and all the branches are "nodes". A hashing system permits efficient and safe verification of massive data structures [5].

Figure 1.1 Merkle tree

Hash trees are utilized in hash-based cryptography. Hash trees are additionally utilized in the IPFS, Btrfs and ZFS record frameworks (to counter information corruption); Dat convention; Apache Wave convention; Git and Mercurial conveyed amendment control frameworks; the Tahoe-LAFS reinforcement framework; Zeronet; the Bitcoin and Ethereum shared organizations; the Certificate Transparency structure; and various NoSQL frameworks like Apache Cassandra, Riak, and Dynamo [6]. Ideas have been made to utilize hash trees in confided in computing frameworks. The underlying Bitcoin execution of Merkle trees by Satoshi Nakamoto applies the pressure step of the hash capacity exorbitantly, which is moderated by utilizing Fast Merkle Trees.

The usage announcing device on the client side is referred to as the savvy metre, which is defenceless against malicious tasks, for example, changing the meter's comprehension. Currently, in the United States, such perusing shifts have resulted in a $6 billion loss. It is indispensable for power utilities to avert malevolent tasks. Additionally, as a result of the influx of astute metres into the brilliant lattice, the pernicious activities become more modern. For instance, a large number of replayed/infused power consumption reports could be vindictively transported away from the control location. If the attacks cannot be distinguished, the control community would be misled and will make incorrect decisions, such as sending a bogus evaluation data to the clients [7]. As a result, it is critical to develop a verification

strategy for identifying replayed/infused messages. Likewise, a clever metre is only equipped with constrained properties, for example, a calculation-required microchip, a small amount of memory, and a low computational limit. Regardless, the calculation overhead is significant for the astute metre. For example, the underlying organisations of Ontario's high-level metering foundation support metre readings every 5–60 minutes. The coming era of intelligent metres aims to reduce these time periods to 1 minute or even less. As such, the created confirmation strategy should minimise the calculation overhead on the savvy metres.

The Merkle Tree keeps up the honesty of the information. Assuming any single detail of exchanges or request of the exchange's changes, these progressions reflected in the hash of that exchange. This switch would course up the Merkle Tree to the Merkle Root, changing the estimation of the Merkle root and along these lines negating the square. So everybody can see that Merkle tree considers a fast and straightforward trial of whether a particular exchange is remembered for the set or not. Merkle tree is a central piece of cloud computing innovation. It is a numerical information structure made out of hashes of various squares of information, and which fills in as an outline of the relative multitude of exchanges in a square. It likewise takes into consideration proficient and secure check of substance in an enormous group of information. It additionally assists with checking the consistency and substance of the information. Both Bitcoin and Ethereum use Merkle Trees structure [8].

The "Merkle Tree" is known as the "Hash Tree". Every leaf node represents a predication, and the past non-leaf nodes are unproven assertions. Since merklings are on paired branches, it is essential to have a larger number of branches than normal. In the event that there are no left and right leaf hashes, we will still get the last one. Today, a variety of methodologies and strategies have been proposed to ensure the total cloud security and inviolability of customer data by co-ops. The terms 'expansion of confidentiality' means increased or decreased transparency, and 'increased or decreased confidentiality' often means improved or decreased resistance to intentional or unintentional tampering, respectively. in the same way, data security was referred to as keeping it out of the hands of the person. as long as we are using appropriate technologies to secure our privacy, including encryption and access controls, protection is equivalent. Simplicity for cloud clients is said to be accessibility, which gives them the ability to use the cloud. In previous work, the technique of intermediary encryption is known as information scrambling.

LITERATURE REVIEW

Khalid et al. [9], (2020) proposed the recently arose cloud computing innovation can give a safe technique to offer verification in (Wireless Sensor Network) WSN and IoT (Internet of Things), due to its cryptographic credits and decentralized property. Cloud computing convention was proposed for trading digital currency bitcoin. The cloud computing highlights can be extemporized dependability, unforgeability, unwavering quality, and adaptation to non-critical failure make a cloud computing strategy is an incredible methodology for verification. Cloud computing gives fuse of keen agreements that gives access control strategy for conveying gadgets.

Tian et al. [10], (2020). Explained as per the creator, they are quick to utilize a key management approach for sensor network dependent on cloud computing innovation. Cloud computing innovation can be conquered numerous limits of the conventional key management framework. Cloud computing has numerous benefits like decentralization, energy utilization, temper sealing, and arrangement. They proposed cloud computing-based key management which has less reliance on the base station. This strategy gives dependability, security, and unwavering quality.

Si Han et al.[11], (2019) proposed a novel gathering key management convention for the information partaking in the cloud stockpiling. In secret sharing gathering key management convention (SSGK), utilizes RSA and confirmed mystery sharing to cause the information proprietor to accomplish fine-grained power over the reevaluated information without depending on any outsider. Besides the creator show that convention displays less capacity and computing intricacy. Security instrument in conspire ensures the protection of lattices information in cloud stockpiling. Encryption gets the transmission on the public channel; checked security conspire make the matrices information just got to approved gatherings. The better presentation regarding capacity and calculation make the plan more viable.

RESEARCH METHODOLOGY

The proposed structure called as "Merkle Tree System" is utilized to part encoded substance to little bits of articles (less or equivalent to 1 MB) and confirm the fulfillment. With all the current plan including, Graph Chain, Tree organization, we can see the 10,000 foot view of a private stockpiling system now.

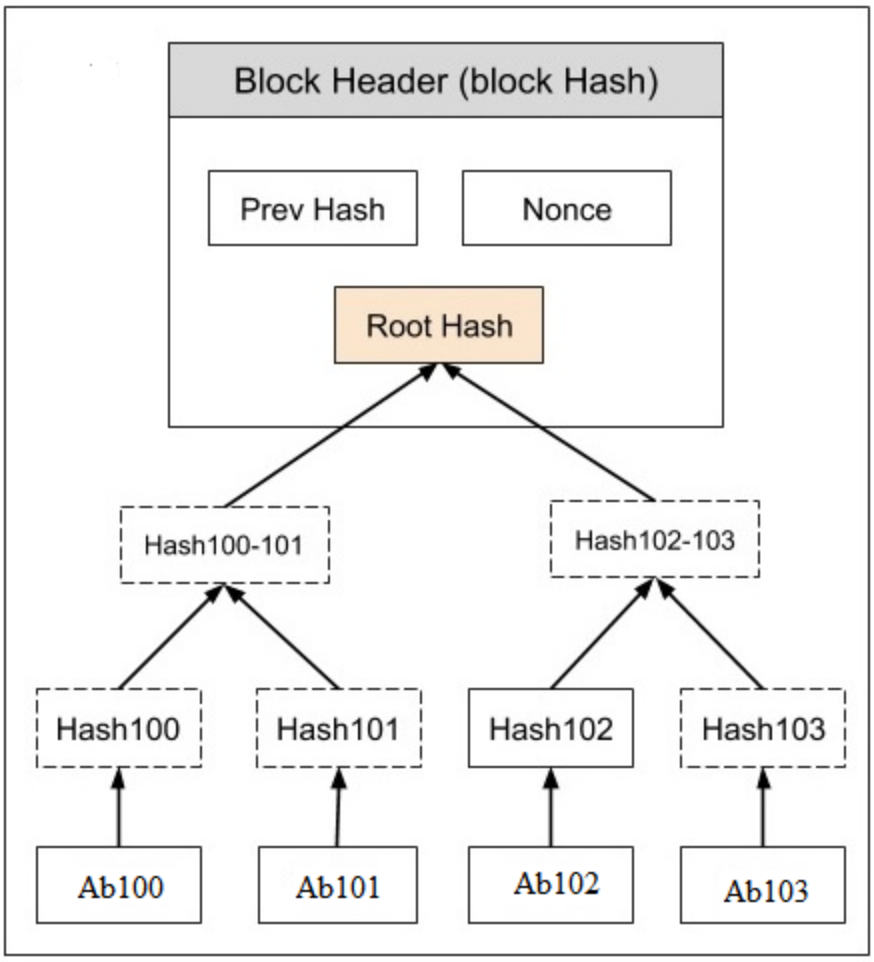

Figure 1.2 System Architecture

Client will actually want to begin utilizing MTFS by producing a public/private key pair. To store a document on the organization, the substance should be scrambled with the client's public key first. After the code text produced, a capacity contract between the client and an asset donor will be marked, and the agreement is recorded in cloud computing. MTFS will have the scrambled substance for the client text to cloud stockpiling.

The above design is the most widely recognized and straightforward type of a Merkle tree, i.e., Binary Merkle Tree. There are four exchanges in a square: Ab1, Ab2, Ab3, and Ab4. Here you can see, there is a top hash which is the hash of the whole tree, known as the Root Hash, or the Merkle Root. Every one of these is over and over hashed, and put away in each leaf hub, bringing about Hash 0, 1, 2, and 3. Successive sets of leaf hubs are then summed up in a parent hub by hashing Hash0 and Hash1, bringing about Hash01, and independently hashing Hash2 and Hash3, bringing about Hash23. The two hashes

(Hash01 and Hash23) are then hashed again to create the Root Hash or the Merkle Root. Like hash records, hash trees are a side product of hash chains. Since leaf hubs is a synonym for quantity, it is needed to compute different hash values relative to the logarithm in order to show that the leaf is a given. Hash trees can be used to obtain, retain, and store any details. They will help to guarantee that various friends in a dispersed organisation get facts are obtained with total integrity, as well as to guard against inadvertent lying and forgery.

Tree network development The tree network is the foundation of MTFS system. It interlinks the conveyed system's assets, assumes the part of correspondence foundation and carries out self administration. Every hub in the organization ought to be a worker with public IP address and open got to port. The principal hub in a tree network becomes root hub, whose bunch identifier (ID) is a vacant string. Any hub in the tree can have up to two youngsters hubs. After the foundation of an association, the parent hub doles out the kid hub a gathering ID. Note that, during the foundation of an association with the parent hub, the kid hub itself can be a parent hub for different hubs.

1) Open branches: The tree network is utilized for message broadcasting; the data of open branches is spreading and synchronized among hubs by message broadcasting. Every hub keeps a duplicate of data posting accessible branches inside the entire tree organization. Two message conventions are characterized for the open branches data management: AVAILABLE BRANCHES and DISCARDED BRANCHES. The message AVAILABLE BRANCHES is utilized to declare new open association focuses to the organization that new hubs can join the organization. Another message DISCARDED BRANCHES shows that current branch as of now acknowledges a kid hub association, so the hub's data will be taken out from the rundown of universally accessible open branches. At the point when a kid hub figured out how to associate with a parent hub, the parent hub sends a DISCARDED BRANCHES message to arrange because of its two accessible branches taken. In the mean time, the interfacing youngster hub will communicate something specific AVAILABLE BRANCHES after its parent hub relegates it a gathering ID, telling the organization another two branches are open.

2) Group Naming: A hub interfaces with a parent hub; there may be two branches accessible for connecting. A parallel tree has a left and a correct branch. At the point when a branch is appended, the parent will reaction a GROUP ID message to affirm the authority bunch ID of the recently joined hub. This activity is to forestall a hub to interface with a tree limb which is as of now taken by other hub. In such a case, the parent will power to separate the copy association.

With, the record proprietor can scramble the document content into a code text and a container. The code text is the length of the substance, however the case is typically short. The container can be re-encoded with sender's private key and beneficiary's public key. When the collector acknowledges the re-scramble container allowed by sender, the beneficiary can undoubtedly decode the code text with the new case and their own private key.

RESULTS & DISCUSSION.

Figure 1.3 shows that the complete preparing time which is contrasted and Secure Hash Algorithm (SHA256) proposed Merkle Tree System (MTS). From the outcomes the proposed MTS shows that the base preparing time when contrasted and SHA.

Table 1 Processing Time

File Size (KB)	Time (sec)	
	Proposed MTS	**Existing SHA**
40	800	1000
80	900	1100
120	1200	1400
160	1250	1450

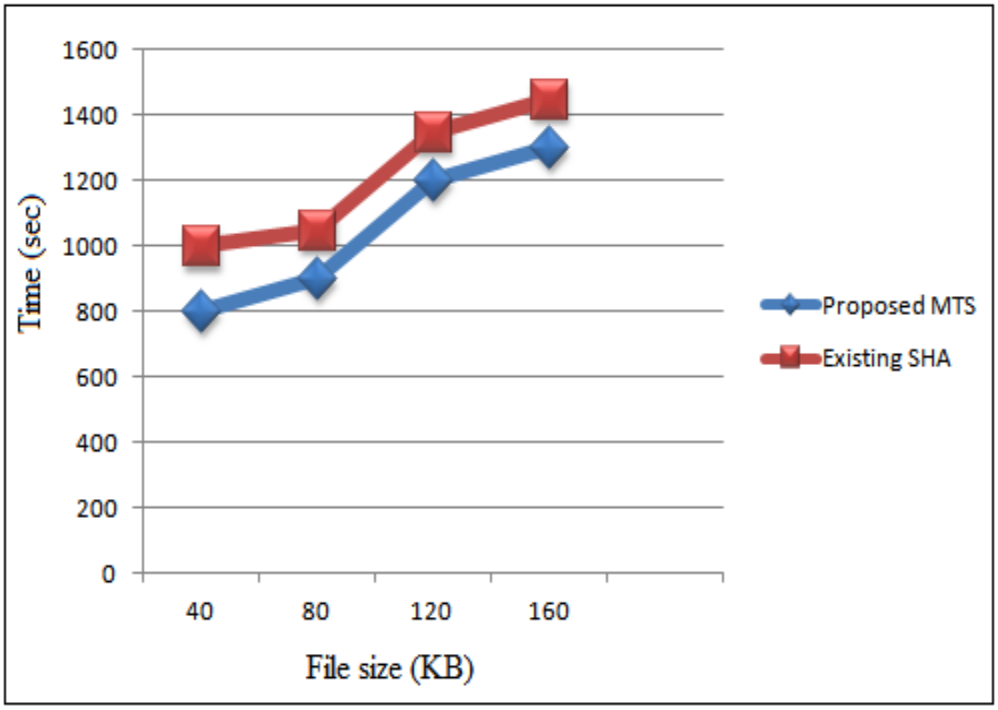

Figure 1.3 Processing time

Figure 1.4 shows that the Data Recovery Time which is contrasted and Secure Hash Algorithm (SHA256 and proposed Merkle Hash Tree (MTS).From the outcomes the proposed MTS shows that the data recovery time when contrasted and SHA.

Table 2 Data Recovery Time

Data Size (%)	Data Recovery Time (sec)	
	Proposed MTS	Existing SHA
40	50	90
80	60	150
120	55	400
160	65	550

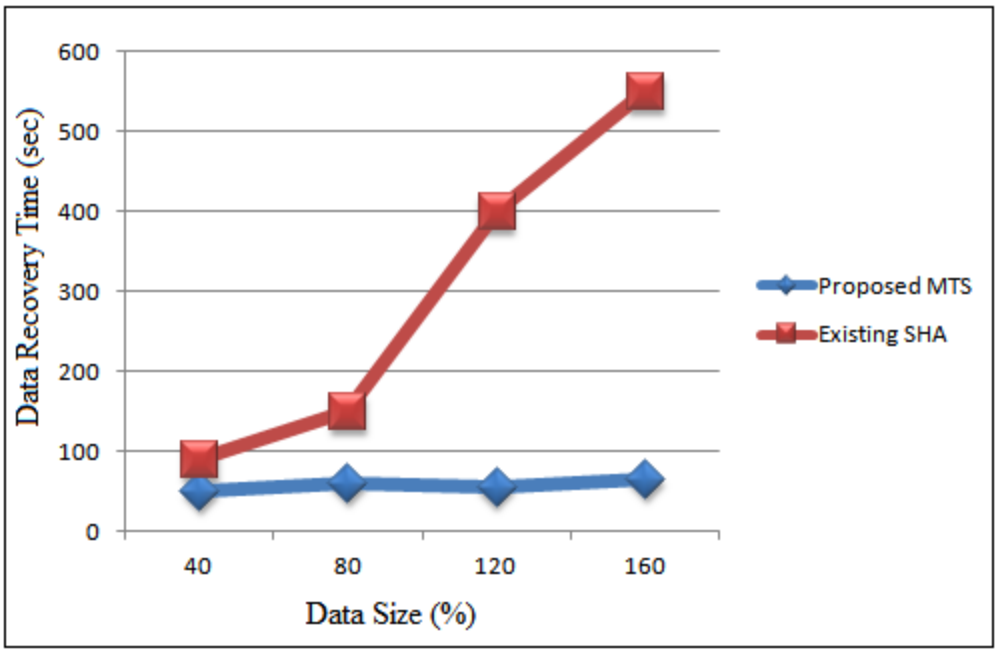

Figure 1.4 Data Recovery Time

CONCLUSION

From the outcomes the proposed MTS shows that the base Data Recovery Timewhen contrasted and SIn this paper, we have proposed MTS and talked about its numerous perspectives from the innovation necessities to the plan and execution with significant spotlight on cryptography, cloud computing, organization and capacity construction. In view of a novel plan we assembling a safe private document stockpiling system utilizing existing advances: A tree network is utilized as a spine network for capacity with elite telecom, where just workers with public IP delivers are permitted to join as hubs to dodge the issues related with firewall and entryway NAT in accomplishing better. Additionally PRE encryption is acquainted with re-scramble the decoded container without altering existing code messages. Base on these segment advancements, the mapping of capacity is planned, and MTS can give a superior answer for a protected private document stockpiling administration dependent on the cloud computing technology.

REFERENCES

1) Y. Cao, Z. Sun, N. Wang, M. Riaz, H. Cruickshank, and X. Liu, "Geographic-based spray-and-relay (gsar): An efficient routing scheme for dtns," IEEE Transactions on Vehicular Technology, vol. 64, no. 4, pp. 1548–1564, April 2015.
2) Jinan Shen,Xuejian Deng, ZhenwuXu, Multi-security-level cloud storage system based on Improved Proxy re-encryption,https://doi.org/10.1186/s13638-019-1614-y, springer, 2019.
3) ShengminXu, Guomin Yang, Yi Muc, Ximeng Liu, A secure IoT cloud storage system with finegrained access control and decryption key exposure resistance, Future Generation Computer Systems 97, Elsevier, (2019) 284–294.
4) O. Zakaria, A.-H. A. Hashim, and W. H. Hassan, "An efficient scalable batch-rekeying scheme for secure multicast communication using multiple logical key trees," International Journal of Computer Science and Network Security (IJCSNS), vol. 14, no. 11, p. 35, 2014.
5) B. Zhou, Q. Chen, P. Xiao, and L. Zhao, "On the spatial error propagation characteristics of cooperative localization in wireless networks," IEEE Transactions on Vehicular Technology, vol. 66, no. 2, pp. 1647– 1658, Feb 2017.
6) Y. Cao, N. Wang, G. Kamel, and Y. J. Kim, "An electric vehicle charging management scheme based on publish/subscribe communication framework," IEEE Systems Journal, vol. PP, no. 99, pp. 1–14, 2015.
7) J. Yang, H. Wang, Z. Ding, Z. Lv, W. Wei, and H. Song, "Local stereo matching based on support weight with motion flow for dynamic scene," IEEE Access, vol. 4, pp. 4840–4847, 2016.
8) D. He, C. Chen, J. Bu, S. Chan, Y. Zhang, and M. Guizani, "Secure service provision in smart grid communications," IEEE Commun. Mag., vol. 50, no. 8, pp. 53–61, Aug. 2012.
9) U. Khalid, Md. Asim, T. Baker, P. C. K. Hung, Md. A. Tariq, and L. Rafferty (2020) A decentralized lightweight cloud computing-based authentication mechanism for IoT systems." Cluster Computing, 23(3): 2067-2087.
10) Y. Tian, Z. Wang, J. Xiong, and J. Ma. (2020) A Cloud computing-Based Secure Key Management Scheme with Trustworthiness in DWSNs." IEEE Transactions on Industrial Informatics.
11) Si Han, Ke Han, and Shouyi hang, A Data Sharing Protocol to Minimize Security and Privacy Risks of Cloud Storage in Big Data Era, IEEE Access, volume 7, 2019.

*e*ISSN 1303-5150