ISSN: 2366-1313

ENABLING FAST PUBLIC AUDITING AND DATA DYNAMICS IN CLOUD SERVICES

[1]Mrs. B.Vijitha,[2] B.Akshitha,[3]G. Meghana,[4]G.Naresh

[1]Assistant Professor, Dept. of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad,

vijitha.boppena@tkrec.in

[2, 3, 4, BTech] Student, Dept. of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad

akshitabodakuntla@gmail.com,gadilameghana@gmail.com,nareshgollamandala675@gail.com

ABSTRACT :

Public auditing enables efficient integrity checks of data will be assigned to cloud servers. We have updated the public auditing for the encrypted data in this specific file. One of our main concerns is the efficiency with which data dynamics—that is, data insertion, alteration, and deletion—are supported. We first determine the factor in the existing auditing schemes most limits data dynamics from the perspective of cost. And we can put forward a novel public auditing scheme that provides a data dynamic that orders of magnitude quicker than the previous procedures. Our novel auditing challenge-response protocol reduces computation cost of the TPA significantly, thus increasing verification speed for the auditing results. The suggested strategy ensures data integrity and privacy against an untrusted cloud while generating minimum computing costs, as demonstrated by performance and security analysis.

Keywords: **Public** auditing, data dynamics, cloud computing.

I INTRODUCTION

Hosting the data in cloud minimizes maintenance requirements, allowing users to easily access their data on cloud servers. However, cloud servers have full check over outsourced data, which raises safety concerns about data integrity .For instance, the cloud may be financially motivated to remove seldom used data, freeing up storage space that may be used, for example, to house additional data-centric apps . Therefore, users need to verify periodically that their data is intact but this has become increasingly onerous due to the ever-growing volume of data being outsourced. Public auditing addresses this issue by utilizing a third- party auditor (TPA) to verify the accuracy. That the data in the cloud on behalf of users.

Public auditing works by dividing a file into many blocks, with each block cooperated with auditing metadata,because only a few blocks need to be tested in order to ensure full integrity, it is particularly helpful for integrity tests. In order to provide data dynamics like block-wise modification, insertion, and deletion, public auditing is necessary as more cloud-backed applications implement dynamic firmware updates. Regretfully, it has been noted that prior strategies provide data dynamics support at a non- trivial cost Specifically, execute an operation on a single block requires a significant volume of auditing metadata associated with other blocks to be updated, delaying integrity checks for dynamic file updates. For example, suppose the data owner uses online collaboration services to share a source code with a group of users.

The data owner splits the code into the fixed number of lines and group them as a set, with each set associated with a tag as auditing metadata. In this instance, adding even a single line to a set's unique location may have an impact on following sets, resulting in additional costs beyond simply recalculating the tag associated with the set. With the set, but also re-computation of the Department of Computer Science and Engineering, Korea University, Seoul, Korea, is home to C. Hahn, H. Kwon, D. Kim, and J. Hurl. Email address: jbhur@korea.ac.krall tags corresponding to the subsequent sets.

For flexible service supply, efficient data dynamics for outsourced data must be considered, since the shared codes are subject to rapid updates. It is also desirable for a TPA to rapidly complete the verification and notify the user of the results. However, according to our measurements, the TPA experiences a delay of approximately 1.0 to 1.6 seconds when verifying the integrity of a file . This lag may not be acceptable because auditing requests can be concentrated within the specific time line Typically a TPA is assigned numerous files, so it would be highly be if it is possible to lower the TPA-side calculation cost for verification.

In this research, we study the element that most restricts data dynamics in terms of cost when departing auditing schemes by revisiting public auditing with data dynamics. We find that auditing metadata mi of block bi in an n-block file is tightly coupled with its index i.We found that cloud misbehavior cannot be prevented without this linkage. Specifically, auditing a file involves (bi,mi) pairs in response to a random selection of indices i as an auditing request. The use of pairs that don't correspond to these indices leads to verification failure.

Unfortunately, this coupling is inefficient in terms of data dynamics because an operation, sayinginsertion ,at the position requires the replacement of the previous auditing metadata (mi,mi+1,...,mn) with new metadata (m0 i+1,m0 i+2,...,m0n+1).We use a new auditing side information approach to carefully isolate the auditing metadata and index in order to address this problem. Specifically, for each block bi, we generate au- dating metadata and auditing side information (SI), both of which are detached from the index but cryptographically coupled to each other. The SI helps the TPA to detect cloud misbehavior, such as submitting pairs that do not correspond to the indices .At the same time, extremely rapid data dynamics is possible because of the auditing metadata is no longer associated with the index. Our experiment demonstrates that the data dynamics need approximately 7 milliseconds in the proposed scheme, which is orders of magnitude faster than 3 to 13 seconds required for previous schemes.

We summarize that the asymptotic costs for data dynamics for an n-block file. Our novel auditing challenge-response protocol reduces the computation cost to the TPA significantly. Specifically, the TPA-side computation cost with respect to verification is a constant number of pairings and exponentiations in a cyclic group, while prior works require those operations linearly with the number of challenged blocks.

II. LITERATURE SURVEY

1.Cong Wang, Sherman S.M. Chow, Qian Wang, KuiRen, and Wenjing Lou (2013) published "Efficient and Secure Dynamic Auditing Protocol for Data Storage in Cloud Computing".

This paper proposes an efficient and secure dynamic auditing protocol (EFS) for cloud storage. It focuses on achieving both data integrity and privacy. EFS supports secure and efficient dynamic operations on outsourced data, making it suitable for cloud environments.

2.GiuseppeAteniese, Randal Burns, Reza Curtmola, Joseph Herring, Lea Kissner, Zachary Peterson, and Dawn Song, "Dynamic Provable Data Possession" (2007).

The paper introduces the concept of dynamic provable data possession (DPDP), where a client can efficiently verify the integrity of data stored at an untrusted server. DPDP allows for dynamic updates to stored data while maintaining efficient verification.

3."Toward Efficient Data Possession Verification in the Cloud" by M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan (2008).

This paper presents a mechanism for efficient data possession verification in cloud storage systems. It proposes the use of Merkle hash trees to provide fast and secure verification of data integrity, making it suitable for dynamic data updates.

4."Efficient Algorithms for Enforcing SLAs in Cloud Storage" by D. Agrawal, A. E. Abbadi, and S. D. C. di Vimercati (2011):

The paper discusses efficient algorithms for enforcing service-level agreements (SLAs) in cloud storage environments. It addresses dynamic data updates and auditing requirements to ensure compliance with SLAs while minimizing overhead.

5."Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing" by Cong Wang, Sherman S.M. Chow, Qian Wang, KuiRen, and Wenjing Lou (2009):

This paper proposes a scheme for enabling public verifiability and data dynamics in cloud storage security. It introduces techniques for efficiently updating and verifying data integrity proofs in dynamic cloud environments.

6."Towards Secure and Dependable Storage Services in Cloud Computing" by Kan Yang, XiaohuaJia, KuiRen, and Bo Zhang (2009)

The paper addresses security and dependability issues in cloud storage services. It discusses techniques for ensuring data integrity, availability, and confidentiality in dynamic cloud environments.

III SYSTEM ANALYSIS

EXISTING SYSTEM

As a hardware-aided solution, auditing metadata could be securely computed on untrusted cloud servers using trusted execution environments like Intel Software Guard Extensions (SGX) In this approach, the computation of auditing metadata can be securely performed with a single machine in the cloud as long as the cloud server is SGX-enabled. However, the workload remains the same in terms of computation. Users would be reluctant to use computationally intensive auditing because cloud computing vendors typically charge for resource usage with pay-as- you-go pricing therefore, an important step toward the more widespread use of auditing is to establish cost- effective data dynamics.

Traditional Systems may not employ block level division and instead might process entire file as a unit during verification .This leads to slow verification .Hashing may be applied to entire files than individual blocks. When modifications occurs, only the affected block need to be recomputed and rehashed .Therefore there are many limitations in exist auditing which does not support data dynamics from cost perspective and consumes more time.

Limitations of Existing System

- ✓ High computational cost
- ✓ Limited support for data dynamics.
- ✓ More verification time

PROPOSED SYSTEM

we present a public auditing scheme designed to meet the aforementioned design goals. After introducing the preliminaries, we provide an overview of the proposed scheme. We then describe the construction of our main scheme and outline how it works in supporting dynamic operations. The proposed scheme consists of the three phases:

Data upload:

This phase is run mostly by the user. The user generates public and secret parameters He divides a file into multiple data blocks before storing it in the cloud. To ensure data confidentiality, the user needs to encrypt the data blocks. To enable the TPA to audit without exposing the key to it, the user computes auditing metadata that feature homomorphic hash properties. The user sends encrypted data blocks and the corresponding auditing metadata to the cloud. He then deletes the data blocks and the auditing metadata from the local storage.

Data update:

In this phase, we assume that the user downloaded some files of interest in advance. If he finds that some blocks of a file need to be updated (e.g., block modification, insertion, and deletion), he erypncts the updated block and generates new auditing data corresponding to the updated block1. Then, he uploads this to the cloud.

Data auditing:

This phase is run interactively between the user, the TPA, and the cloud. When the user wants to check the integrity of the data stored in the cloud, the user sends an auditing request to the TPA. Upon receiving the auditing request from the user, the TPA generates and sends a challenge message to the cloud server. The cloud server derives proof from the stored data and sends it back to the TPA. Finally, upon receiving proof from the cloud, the TPA verifies the correctness of the proof. If the verification succeeds ,it indicates that the integrity of the file is intact; otherwise, the file is compromised. We note that this phase can run without the user: the TPA periodically audit data of its choice by interacting with the cloud.

Proposed system Advantages:

- ✓ The proposed system is a privacy-preserving public auditing mechanism for shared data in the cloud.
- ✓ The user sends encrypted data blocks and the corresponding auditing metadata to the cloud. He then deletes the data blocks and the auditing metadata from the local storage.

IV IMPLEMENTATION

Architecture:

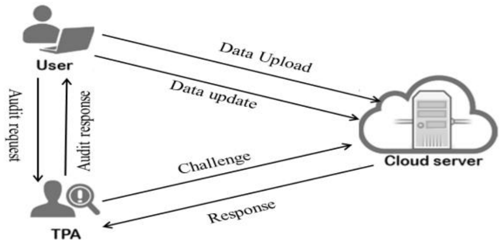


Fig-1. Architectures of the system model

Architectural Design:

An application can be divided into three primary logical components using the Model-View- Controller (MVC) architectural pattern: the model, the view, and the controller. Each of these parts is designed to manage particular application development tasks. One of the most widely used web development frameworks that is industry standard for creating scalable and flexible projects is MVC.

MVC Components

Following are the components of MVC –

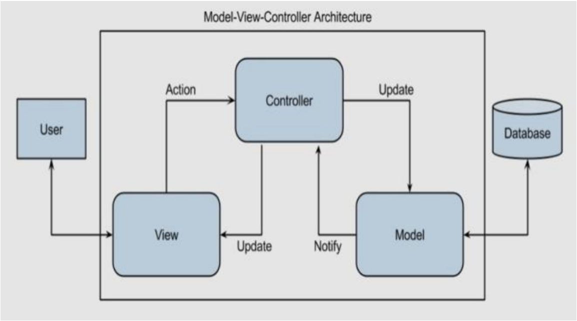


Fig-2: Model View Controller Architecture

Model

All of the user's data-related logic is mapped to the Model component. This might be any extra data relevant to business logic, or it can represent the data being transmitted between the View and Controller components. For instance, a customer object will access customer data from the database, modify it, and then either render data using it or return it to the updated database.

View

The application's UI functionality is all implemented using the View component. For instance, every UI element that the end user interacts with, including text fields and dropdown menus, will be present on the Customer view.

Controller

Controllers conduct all business logic and incoming requests, use the Model component to manipulate data, and communicate with the Views to render the final product by acting as an interface between them. For instance, the Customer controller will use the Customer Model to update the database and manage all interactions and inputs from Customer View.

The same controller will be used to view the customer data.

MODULES

Data Owner

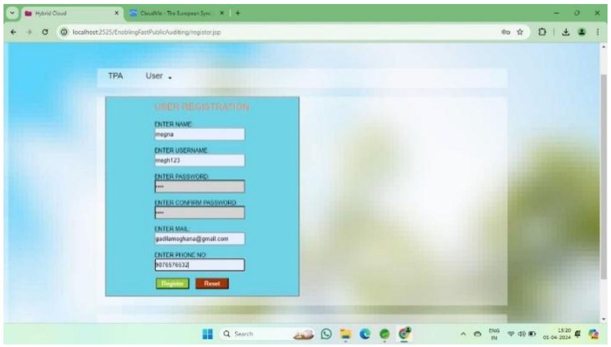
- Registration
- Login
- upload file: Split files into blocks generate hash signature encrypt file blocks generate block index upload to cloud update file block get file blocks from the cloud decrypt file blockscheck integrity verification
- update file block: Generate hash signature encrypt file blocks update file index the update file in the cloud
- Delete file block: Get file blocks from the cloud decrypt file blocks check integrity verification delete file block update file index
- Logout

TPA

- Login
- View files: View file blocks generate audit request perform audit challenge submit audit challenge
- Logout

V RESULT AND DISCUSSION

User Registration Page:



localhost:2525/EnablingFastPublicAuditing/register.jsp

USER REGISTRATION

ENTER NAME: megna

ENTER USERNAME: megh123

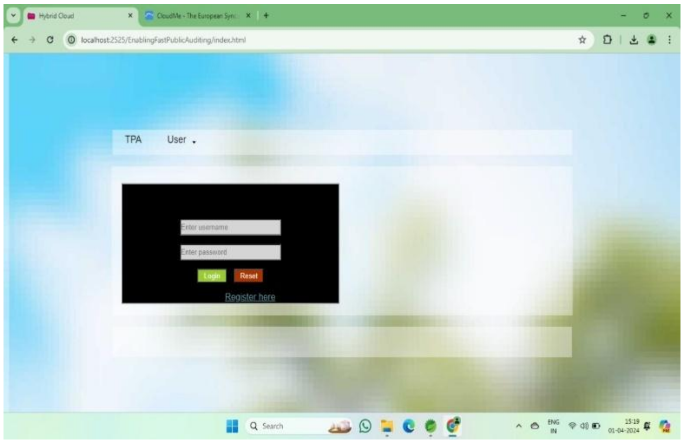
ENTER PASSWORD: ****

ENTER CONFIRM PASSWORD: ****

ENTER MAIL: gadilamoghana@gmail.com

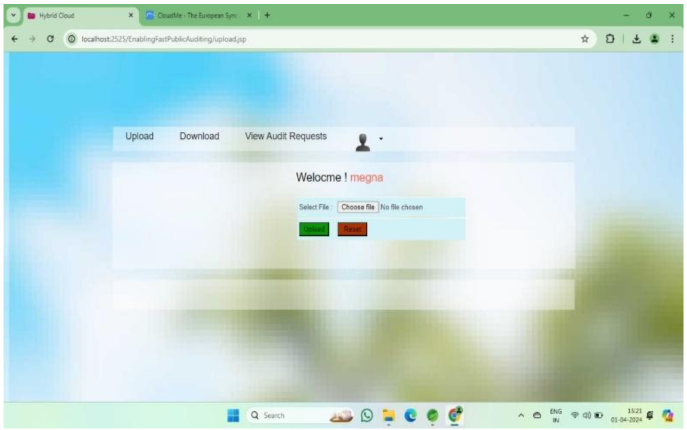
ENTER PHONE NO: 9876576532

User Login Page:



localhost:2525/EnablingFastPublicAuditing/index.html

File Upload Page:

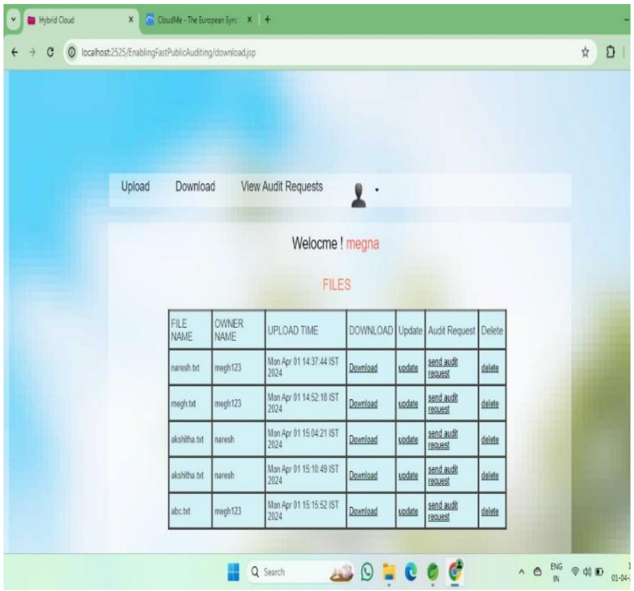


localhost:2525/EnablingFastPublicAuditing/upload.jsp

Welcome ! megna

Select File: Choose file No file chosen

View files:

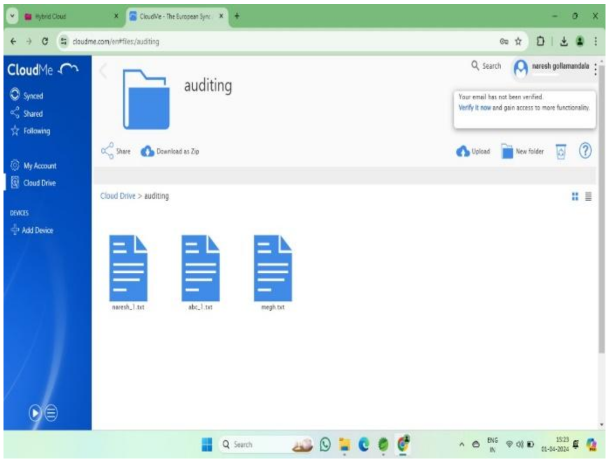


Welocme ! megna

FILES

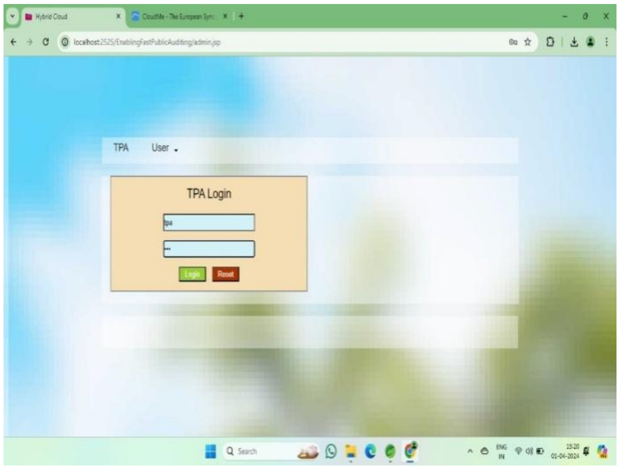
FILE NAME	OWNER NAME	UPLOAD TIME	DOWNLOAD	Update	Audit Request	Delete
naresh.txt	megh123	Mon Apr 01 14:37:44 IST 2024	Download	update	send audit request	delete
megh.txt	megh123	Mon Apr 01 14:52:18 IST 2024	Download	update	send audit request	delete
akshitha.txt	naresh	Mon Apr 01 15:04:21 IST 2024	Download	update	send audit request	delete
akshitha.txt	naresh	Mon Apr 01 15:10:49 IST 2024	Download	update	send audit request	delete
abc.txt	megh123	Mon Apr 01 15:15:52 IST 2024	Download	update	send audit request	delete

Uploaded files in colud:



Your email has not been verified.
Verify it now and gain access to more functionality.

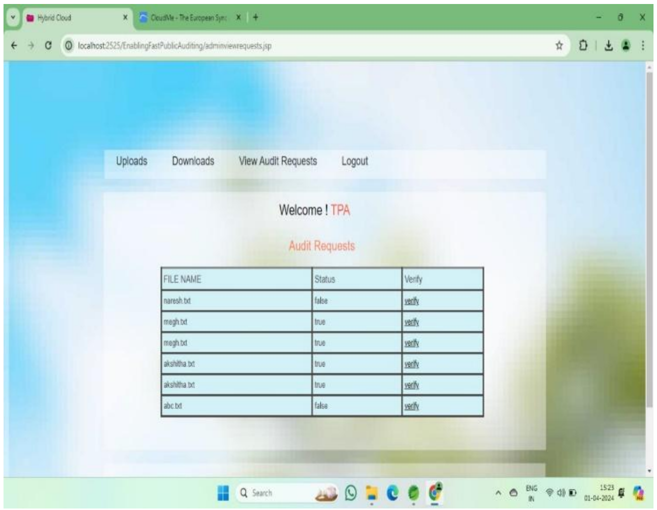
TPA Login Page:



TPA Login

tpa

View Audit Requests:

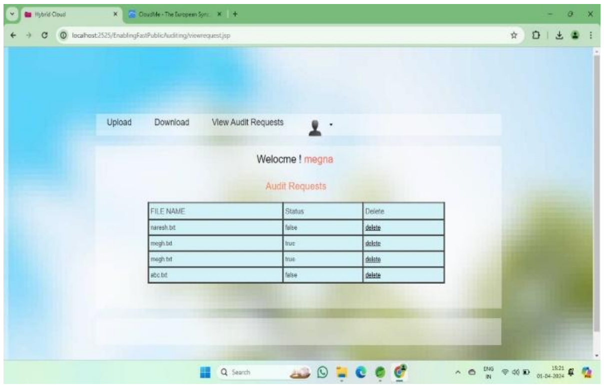


Welcome ! TPA

Audit Requests

FILE NAME	Status	Verify
naresh.txt	false	verify
megh.txt	true	verify
megh.txt	true	verify
akshitha.txt	true	verify
akshitha.txt	true	verify
abc.txt	false	verify

View Audit Status:



Welocme ! megna

Audit Requests

FILE NAME	Status	Delete
naresh.txt	false	delete
megh.txt	true	delete
megh.txt	true	delete
abc.txt	false	delete

VI CONCLUSION

We propose a public auditing method for encrypted data in this research that enables incredibly rapid data dynamics. The suggested structure allows for data dynamics regardless of the number of blocks at a fixed cost. The auditing results may be verified much faster thanks to our auditing challenge-response system, which needs a fixed amount of pairings and

ISSN: 2366-1313

exponentiations. Data confidentiality and integrity are ensured by the plan suggested in comparison to the cloud server. Because of the homomorphic hash function, the TPA can verify the accuracy of the proof during the auditing process without having to decode it or reveal the key. Analysis of security and performance reveals that the suggested technique requires little additional analyzing while upholding data integrity and privacy.

FUTURE ENHANCEMENT

In the realm of enabling fast public auditing and dynamic data management within cloud services, future enhancements are poised to revolutionize the landscape by addressing existing challenges and pushing the boundaries of efficiency, scalability, and security. One significant avenue for advancement lies in enhancing the efficiency and scalability of auditing mechanisms. Novel algorithms and optimizations can streamline auditing processes, reducing computational overhead and enabling seamless scalability to handle increasingly large and complex datasets.

Moreover, the dynamic nature of data in cloud environments necessitates innovative strategies for managing data updates, deletions, and migrations while ensuring data integrity and auditability. Future research may focus on developing automated auditing frameworks that continuously monitor cloud storage systems, leveraging machine learning and anomaly detection techniques to proactively identify and mitigate security threats and compliance violations. Additionally, the integration of blockchain technology holds promise for enhancing the transparency, immutability, and auditability of cloud storage systems, providing a decentralized and tamper-proof ledger for recording audit trails. Privacy-preserving auditing methods, dynamic access control mechanisms, and real-time monitoring and response capabilities are also ripe areas for exploration, offering enhanced data security, compliance, and user trust in cloud services.

Ultimately, future enhancements in this domain will empower organizations to leverage cloud services with confidence, knowing that their data remains secure, auditable, and adaptable to dynamic business needs.

Develop novel algorithms and techniques to further improve the efficiency and scalability of auditing mechanisms. This includes optimizing data structures, reducing computational overhead, and parallelizing auditing processes to handle large-scale data sets efficiently.

Investigate methods for conducting auditing operations while preserving the privacy of sensitive information. This could involve techniques such as homomorphic encryption, zero-knowledge proofs, and differential privacy to enable secure auditing without exposing data contents.

Integrate dynamic access control mechanisms with auditing frameworks to enforce fine-grained access policies based on user roles, privileges, and data sensitivity. This enhances data security by ensuring that only authorized users can access and modify data, while audit logs provide transparency and accountability.

In the future, making audits faster and handling changing data better in cloud services will be improved. This means finding ways to quickly check data integrity and adapt to changes without slowing things down. We'll see smarter systems that automatically monitor and fix problems, like security threats or changes in regulations. Blockchain technology might also play a bigger role, ensuring that records of audits are secure and tamper-proof. Plus, there will be a focus on keeping data private while still allowing audits, and making sure only the right people can access it. Overall, these improvements will make cloud services safer, more reliable, and easier to use for everyone.

VII REFERENCES

[1] Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Kon- winski, A., ...&Zaharia, M. (2010). A view of cloud computing. Communications of the ACM, 53(4), 50-58.

[2] Ren, K., Wang, C., & Wang, Q. (2012). Security challenges for the public cloud. IEEE Internet Computing, 16(1), 69-73.

[3] Song, D., Shi, E., Fischer, I., & Shankar, U. (2012). Cloud data protection for the masses. Computer, 45(1), 39-45.

[4] Prasadu Peddi (2018), "A STUDY FOR BIG DATA USING DISSEMINATED FUZZY DECISION TREES", ISSN: 2366-1313, Vol 3, issue 2, pp:46-57..

[5] Ateniese, G., Burns, R., Curtmola, R., Herring, J., Kissner, L., Pe- terson, Z., & Song, D. (2007, October). Provable data possession at untrusted stores. In Proceedings of the 14th ACM conference on Computer and communications security (pp. 598-609).

[6] Ateniese, G., Di Pietro, R., Mancini, L. V., &Tsudik, G. (2008, September). Scalable and efficient provable data possession. In Proceedings of the 4th international conference on Security and privacy in communication netowrks (pp. 1-10).

[7] Prasadu Peddi (2015) "A review of the academic achievement of students utilisinglarge-scale data analysis", ISSN: 2057-5688, Vol 7, Issue 1, pp: 28-35.

[8] Shacham, H., & Waters, B. (2008, December). Compact proofs of retrievability.InInternationalconferenceonthe theoryandapplica- tion of cryptology and information security (pp. 90-107). Springer, Berlin, Heidelberg.

[9] Erway, C. C., K¨upc¸¨u, A., Papamanthou, C., &Tamassia, R. (2015). Dynamic provable data possession. ACM Transactions on Informa- tion and System Security (TISSEC), 17(4), 1-29.

[10] Wang, Q., Wang, C., Ren, K., Lou, W., & Li, J. (2010). Enabling public auditability and data dynamics for storage security in cloud computing. IEEE transactions on parallel and distributed systems, 22(5), 847-859.

11] Prasadu Peddi (2023). AI-Driven Multi-Factor Authentication and Dynamic Trust Management for Securing Massive Machine Type Communication in 6G Networks. International Journal of Intelligent Systems and Applications in Engineering, 12(1s), 361–374.

[12] Wang, C., Chow, S. S., Wang, Q., Ren, K., & Lou, W. (2011). Privacy-preserving public auditing for secure cloud storage. IEEE transactions on computers, 62(2), 362-375.

AUTHORS

Mrs. B.Vijitha, Assistant Professor Dept. of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad.

Email: vijitha.boppena@tkrec.in

Miss. B.Akshitha,Dept. of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad.

Email: akshitabodakuntla@gmail.com

Miss. G. Meghana ,Dept. of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad.

Email: gadilameghana@gmail.com

Mr. G.Naresh,Dept. of CSE, Teegala Krishna Reddy Engineering College, Meerpet, Hyderabad.

Email: nareshgollamandala675@gail.com