Improving Security in Cloud Storage: Auditing by Identity, Hidden Data and Secure Sharing

Mr Mohammed Abdul Raqeeb Sher[1], Mr Kaleem Salman[2], Mr Mohammed Nizam Uddin[3],
Mr Mirza Younus Ali Baig[4], Mr Md Anjar Ahsan[5], Mr. Syed Altaf[6]
[1,2,3]*UG Scholar, Lords Institute of Engineering and Technology*
[4,5,6]*Assistant Professor, Lords Institute of Engineering and Technology*

***Abstract*—In the world of web security testing, a crucial task is to identify weaknesses in websites that could allow unauthorised access. This abstract focuses on a specific aspect called "Username Enumeration via Subtly Different Responses." It's a method used by hackers and security experts to figure out valid usernames for a website. Before hackers try to guess passwords or break into a website, they often want to know if a username exists. This technique helps them find out. It works by carefully examining how a website responds when you try different usernames. The key to this technique is spotting small differences in how the website responds. These differences can be things like error messages that look a bit different, how long it takes for the website to respond, or the codes the website sends back. This abstract goes into more detail about how hackers and security experts use these differences to find usernames. It talks about methods like looking at error messages, timing how long it takes for the website to respond, and using computer programs to spot these differences automatically. Understanding this technique is essential for making websites more secure. When we know how hackers find usernames, we can better protect websites from unauthorised access. This abstract is a basic introduction for security experts and testers, helping them learn how to defend websites against this kind of reconnaissance. In summary, "Username Enumeration via Subtly Different Responses" is a way hackers and security experts discover usernames on websites. This abstract explains how it works, stressing the importance of finding and fixing these issues to prevent unauthorised access and data breaches.**

I. INTRODUCTION

The rapid development of communications and networks in recent years has increased data transmission and exchange. Meanwhile, the demand for multimedia, such as video, pictures, and audio, is increasing. Due to this important development, providing information technology (IT) services has become extremely costly to individuals and businesses. In this regard, Cloud computing is an efficient and good environment in terms of providing necessary IT services due to its economic advantages (Tian et al., 2019).

Cloud computing is a good experience with deep implications for changing the way enterprise uses IT. One of the key aspects of this model is that the data is focused or intended for cloud computing. From the viewpoint of users, involving information technology enterprises and individuals, remote data storage in the cloud paradigm in a flexible on-request method brings good advantages such as reduce the load on storage management, overall data access to different geographic positions, and decrease spending on devices, software, maintenance, etc. (Alrabea, 2020). Cloud storage is one of the basic technologies in the cloud paradigm. Many systems discussed it for low cost and high efficiency of cloud storage; therefore cloud data storage will transform data centres into a large scale computing service. Of course, the fast growth of bandwidth to the network combined with trust and flexible connection of the network will make users enjoy high-quality cloud services (Ping et al., 2020). Cloud storage is dissimilar from traditional storage technologies. It affords large storage space for users and access to data through separate geographical locations. In other words, cloud users can easily access external data from any device connected to the network and connected to the cloud model anytime and anywhere

Despite the enormous benefits of the cloud model, there are also security challenges facing users through their use of outsourced data. Since the management

entities of the cloud service provider are separate, the users will relinquish control over their data. Thus, there are many reasons why data correctness on the cloud is at risk. First, cloud computing infrastructures face external and internal threats to data integrity, instances of service unavailable and security breaking of noticeable cloud computing services emerge from one interval to another Moreover, cloud service providers have many incentives to perform dishonestly to the status of cloud computing users concerning their external data.

II. LITERATURE SURVEY

1. Cloud Data Security using Auditing Scheme Anuj Kumar Yadav, M L Garg et al. Cloud computing has emerged as one of the latest computing paradigm and is a growing technology for upcoming years. According to NIST Cloud computing is a model for convenient, on-demand network access to a large pool of computing resources. Resource can be hardware or software resource and this pool of resources can be rapidly provisioned and released with minimum management effort or cloud service provider interaction [1]. In Cloud computing, different types of data and program can be stored at different locations, the cloud data centers and can be accessed whenever required, from anywhere, via different type of devices having internet connection. Due to this method of storing user's data at cloud provider's end, users gets numerous benefits such as, access flexibility, large storage capability, and resilience. In Cloud computing vendor supplies the hardware infrastructure, and the software interacts with the user through a front-end portal.

2 Data Security and Privacy Protection for Cloud Storage: A Survey
Pan yang 1 , naixue xiong2, and jingli ren 1 et.al. The new development trends including Internet of Things (IoT), smart city, enterprises digital transformation and world’s digital economy are at the top of the tide. The continuous growth of data storage pressure drives the rapid development of the entire storage market on account of massive data generated. By providing data storage and management, cloud storage system becomes an indispensable part of the new era. Currently, the governments, enterprises and individual users are actively migrating their data to the cloud. Such a huge amount of data can create magnanimous wealth. However, this increases the possible risk, for instance, unauthorized access, data leakage, sensitive information disclosure and privacy disclosure. Although there are some studies on data security and privacy protection, there is still a lack of systematic surveys on the subject in cloud storage system. In this paper, we make a
comprehensive review of the literatures on data security and privacy issues, data encryption technology, and applicable countermeasures in cloud storage system. Specifically, we first make an overview of cloud storage, including definition, classification, architecture and applications. Secondly, we give a detailed analysis on challenges and requirements of data security and privacy protection in cloud storage system. Thirdly, data encryption technologies and protection methods are summarized.

III. TOOLS USED

3.1 FUNCTIONAL REQUIREMENTS

Admin: Conduct a comprehensive analysis of existing cloud storage auditing systems to identify security Vulnerabilities and areas for improvement related to identity management, detection of hidden data, and secure sharing. Initiator: Collaborate with the project team to design the architecture and functionalities of the initiator module, ensuring alignment with project goals and industry best practices.

Responder: Develop robust identity verification mechanisms within the responder module to authenticate users and ensure that only authorized individuals can respond to requests within the cloud storage environment. Implement algorithms and data analysis techniques to detect and address hidden data within the cloud storage system, enhancing transparency and compliance with data privacy regulations.

3.2 NON-FUNCTIONAL REQUIREMENTS

This section elaborates on the functional requirements of the application. The SRS itself can be divided into module, each module having specifications. In order to carry out the project, the following hardware and software is required.

3.2.1. HARDWARE REQUIREMENTS

System : i3
Hard Disk : 40 GB.
Floppy Drive : 1.44 Mb.
Monitor : 15 VGA Colour.
Mouse : Logitech. RAM : 512 Mb.

3.2.2 SOFTWARE REQUIREMENTS

Technology	: JDBC
Web Server	: Tomcat 7.0
Client Side Technologies	: HTML, CSS, JS
Server Side Technologies	: Servlets, JSP
Data Base Server	: MySQL
Editor	: Net beans

IV. METHODOLOGY

The proposed methodology focuses on developing a secure, identity-based data auditing system that ensures data integrity, supports hidden data protection, and enables secure data sharing in a cloud storage environment. The methodology comprises the following stages:

4.1. Requirement Specification

The system requirements were identified and categorized into functional and non-functional specifications. The functional requirements involve roles such as Admin, Initiator, and Responder, each with specific responsibilities related to auditing, verification, and secure sharing. Non-functional requirements emphasized system security, availability, and performance.

4.2. Technology Stack

The system was built using a widely adopted technology stack to ensure scalability and ease of deployment:

- Frontend Technologies: HTML, CSS, JavaScript

Backend: Java (Servlets, JSP) with JDBC

- Database: MySQL
- Web Server: Apache Tomcat 7.0
- IDE: NetBeans

This stack allows for the development of a modular and secure web-based application with real-time data interaction.

4.3. System Design

Comprehensive modeling techniques were employed to design the architecture:

- UML Diagrams: Use Case, Class, Sequence, Activity, Component, and Deployment diagrams were used to illustrate system behavior and structure.
- ER Diagram: Represented the logical relationships among entities in the database.

Data Flow Diagram (DFD): Described the flow of data within the system to ensure clarity in process modeling.

These diagrams facilitated effective communication of the system architecture and design choices.

4.4. Module Development

The implementation was divided into several modules:

- User Module: Handles user profile creation, login authentication, and data access.
- PKG (Initiator): Initiates secure data upload and sharing protocols.
- Sanitizer (Responder): Validates the integrity of data and detects hidden information.
- TPA (Third Party Auditor): Performs data integrity auditing without accessing the actual content.
- Cloud Module: Manages data storage and interaction with users and auditors.

Each module was designed to operate independently while integrating securely with other components.

4.5. Security Implementation

The core of the methodology involves a privacy-preserving auditing scheme:

- Identity-Based Auditing: Enables secure verification using identity attributes without exposing user data.
- Sensitive Data Hiding: Allows shared data to be partially masked based on sensitivity levels.

- Key Exposure Resilience: Utilizes bilinear pairing techniques to ensure both forward and backward security in the event of key leakage.
- Batch Auditing Support: Enhances efficiency by allowing simultaneous auditing of multiple data blocks.

4.6. Validation and Testing

The system was validated through both analytical proofs and experimental tests. The security model was demonstrated using the computational Diffie–Hellman assumption. Experimental analysis confirmed the system's efficiency in handling cloud auditing tasks under various conditions.

V. RESULTS AND DISCUSSION

The implementation of the identity-based auditing scheme was evaluated through theoretical analysis and practical system deployment using Java, MySQL, and associated web technologies. The following observations were made:

5.1. Data Integrity and Audit Efficiency

The system successfully performed remote data integrity verification without requiring users to download or access the entire dataset. Using identity-based cryptography and bilinear pairing, the system ensured that:

- Auditing could be securely executed even after secret key exposure.
- Sensitive parts of the shared data remained hidden, satisfying privacy requirements.
- The data possession was validated with minimal computation and communication overhead.

5.2. Key Exposure Resilience

A significant improvement was observed in terms of key exposure handling. By incorporating forward and backward security, the system preserved audit security even when the auditing secret key was compromised. This ensures continuous trust in the audit process.

5.3. Batch Auditing Support

To reduce the workload of the Third Party Auditor (TPA), the system incorporated batch auditing, which allowed multiple files or data blocks to be verified simultaneously. This significantly reduced computation time and increased scalability in real-world applications.

5.4. System Usability and Performance

Using common tools such as HTML, CSS, Java, and MySQL made the system easy to deploy and test in a standard environment. The use of modular architecture and clearly defined user roles (Admin, Initiator, Responder, and TPA) facilitated better security policy enforcement and ease of use.

5.5. Limitations

Despite the success of the implementation, certain limitations remain:

- The current system depends on trusted cloud providers for data storage and partial computation.
- It assumes a secure initial key distribution and registration process.

Future enhancements could involve integrating blockchain-based identity management or homomorphic encryption to further strengthen the security and decentralization.

VI. CONCLUSION

In this project, we proposed an identity-based data integrity auditing scheme for secure cloud storage, which supports data sharing with sensitive information hiding. In our scheme, the file stored in the cloud can be shared and used by others on the condition that the sensitive information of the file is protected. Besides, the remote data integrity auditing is still able to be efficiently executed. The security proof and the experimental analysis demonstrate that the proposed scheme achieves desirable security and efficiency.

The need to preserve the integrity of data stored in the cloud rises with the growing demand for storage-as-a-service offering of cloud. So, cloud storage auditing schemes are designed to verify the possession of cloud data but there are critical issues in these auditing schemes. This paper studies the auditing secret key exposure problem in Identity-based cloud storage auditing schemes. Since exposure of secret key is undetectable, a better way to handle the key exposure problem is to minimize the damage caused by the exposed key. An Identity-basedstrong key-exposure resilient cloud auditing scheme using bilinear pairing

is designed and implemented. The proposed scheme preserves the security of cloud auditing both before and after the key exposure by forward and backward security mechanism. Batch auditing is also incorporated into the scheme to ease the workload of the auditor. The scheme is provably secure using the computational Diffie–Hellman assumption in the random oracle model. Experimental results show that the proposed scheme is efficient in auditing the data block.

User Education and Awareness:
Educate users about best practices for safeguarding their own information, such as setting strong passwords, being cautious about sharing personal data, and understanding the platform's privacy features. Encourage users to report any suspicious activities promptly.

Compliance with Regulations:
Adhere to relevant data protection and privacy regulations, such as GDPR, CCPA, or other regional laws. Ensure the platform's policies and practices align with the legal requirements to protect user rights and privacy.

Regular Data Purging and Retention Policies:
Enforce data retention policies to delete unnecessary user data after a specified period. This helps reduce the potential harm in case of a security breach and ensures that data is only stored for as long as necessary.

VII. FUTURE SCOPE

Safeguarding user information in contextual social networks is a critical concern given the increasing amount of personal data shared and the potential privacy risks associated with it. Here are some future-focused strategies and considerations for enhancing user information protection in contextual social networks:

1. Privacy by Design and Default:
Implementing privacy features into the design and architecture of social networks from the outset. Privacy should be the default setting, and users should have granular control over their data sharing preferences.

2. Advanced Encryption and Security Measures:
Enhancing encryption protocols and adopting state-of-the-art security measures to protect user data both during storage and transmission within the social network platform.

3. Decentralized Identity and Authentication:
Implementing decentralized identity systems, like blockchain-based solutions, to allow users to control and manage their identity and personal data independently from the social network, enhancing privacy and security.

4. Collaboration and Information Sharing:
Encouraging collaboration and sharing of best practices within the industry to collectively work towards developing more secure and privacy-focused solutions for contextual social networks.

The future of safeguarding user information in contextual social networks will involve a multidimensional approach, incorporating technology advancements, regulatory compliance, user empowerment, and a commitment to respecting individual privacy. Balancing the benefits of data sharing with the imperative to protect user privacy will be a central challenge in the evolving landscape of social networking.

BIBLIOGRAPHY REFERENCES

[1] Prescient & Strategic Intelligence, Global Enterprise Data Storage, Prescient & Strategic Intelligence, Noida,India. https://www.psmarketresearch.com/marketanalysis/enterprisedata-storage-market.

[2] A. D. Rayome, “69% of enterprises moving business-critical applications to the cloud,” 2019, https://www.techrepublic.com/article/69-of -enterprises-moving-business-criticalapplications-to-the-cloud/. View at: Google Scholar

[3] T.Singleton, “Why are so many enterprises moving to the cloud?” 2018,https://datometry.com/blog/moving-to the-cloud-survey-analysis/.View at: Google Scholar

[4] T. Nikl and R. K. Chintalapudi, “8 common reasons why enterprises migrate to the cloud,” 2018,

https://cloud.google.com/blog/products/storage-data-transfer/8-common-reasons-whyenterprises-migrate-to-the-cloud. View at: Google Scholar

[5] Flexera, “Cloud computing trends,”2019, https://www.rightscale.com/blog/cloud-industryinsights/cloud-compu-ting-trends-201 9-state-cloud-survey. View at: Google Scholar

[6] R.Oskoui, “5 Key cloud security challenges,” 2018, https://www.cdnetworks.com/cloudsecurity /5-key-cloud-security-challenges/4208/. View at: Google Scholar

[7] G. Ateniese, R. Burns, R. Curtmola et al., “Provable data possession at untrusted stores,” in Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS’07),pp. 598–610, Alexandria, VA, USA, November 2007. View at: Publisher Site | Google Scholar

8.A. Juels and B. S. Kaliski, “PORs: proofs of retrievability for large files,” in Proceedings of the 14th ACM Conference on Computer and Communications Security (CCS’07), pp. 584–597, New York, NY, USA, November 2007. View at: Publisher Site | Google Scholar