Biometric-Based Unlocker Framework for Cost-Effective Two-Factor Cloud Authentication

Phani Babu Diyyana[1] and Dr. Pawan Kumar[2]

[1]Research Scholar, Department of Computer Science, Shri Venkateshwara University, Gajraula, UP, INDIA
Email: phanibabu.diyyana@gmail.com
[2]Research Guide, Department of Computer Science, Shri Venkateshwara University, Gajraula, UP, INDIA

Received: 19-10-2024 Revised: 20-11-24 Accepted: 28-12-2024

ABSTRACT
This study presents a comparative evaluation of the proposed Frag Secure framework against state-of-the-art random fragmentation approaches and non-fragmentation strategies. By addressing data loss and security challenges, the Frag Secure framework demonstrates superior performance, achieving enhanced data protection and reliability. The framework highlights a significant improvement in managing fragmented data, ensuring a robust solution to the limitations of existing methods in secure data management systems. The research also introduces a Biometric-Based Unlocker Cloud Framework as an innovative alternative to traditional two-factor authentication methods that rely on costly external hardware devices. By leveraging biometric technology, this framework eliminates concerns about device damage and associated repair costs, providing a seamless and cost-effective authentication mechanism. Performance tests reveal improvement over existing frameworks, offering enhanced security and usability without additional expenses for cloud users. Together, these advancements pave the way for more secure and efficient cloud-based systems.

Keywords: Biometric, Cloud Storage, Data Security, Storage System

1. INTRODUCTION

The rapid evolution of digital systems has brought about significant advancements in data management and security, particularly in cloud computing environments. With the proliferation of sensitive data being stored and accessed remotely, the need for robust and efficient security frameworks has become paramount. Traditional methods, such as two-factor authentication using external hardware devices, have been widely adopted to enhance security. However, these methods come with inherent drawbacks, including high costs, susceptibility to device loss or damage, and the inconvenience of maintenance and replacement. This has driven researchers and industry practitioners to explore innovative solutions that can provide enhanced security while minimizing costs and operational complexities [11-12].

In this context, the proposed Frag Secure framework and Biometric-Based Unlocker Cloud Framework address key challenges in data fragmentation and authentication, respectively. The Frag Secure framework is designed to optimize data security by preventing data loss while improving reliability, outperforming existing random fragmentation and non-fragmentation approaches. This ensures the protection of sensitive information in cloud environments without compromising performance [13].

On the other hand, the Biometric-Based Unlocker Cloud Framework introduces a novel approach to user authentication by replacing external hardware with biometric technology. This framework not only reduces costs but also enhances usability and eliminates the risk of hardware failure or loss. By leveraging biometric data, it ensures secure and seamless access to cloud systems, making it a practical and efficient alternative to traditional authentication methods. Together, these frameworks contribute to the advancement of secure and cost-effective cloud computing solutions, addressing the growing demand for reliable and user-friendly security mechanisms [14].

2. LITERATURE REVIEW

The rapid advancements in cloud computing have necessitated the development of secure and efficient data management frameworks. Smith et al. [1] proposed a random fragmentation approach to enhance data security, demonstrating its effectiveness in reducing the risks of data breaches. Similarly, Johnson and Lee [2] conducted a comparative analysis of non-fragmentation and fragmentation strategies, emphasizing the superior security benefits of fragmented approaches. However, these studies highlighted limitations such as scalability and performance metrics, which remain critical for real-world applications. Addressing these

issues, Patel and Singh [3] introduced a biometric-based authentication method, showcasing its potential to replace traditional authentication mechanisms with more secure and user-friendly solutions.

Despite the promising results, challenges persist in implementing these solutions on a larger scale. Kumar et al. [4] explored hardware-based two-factor authentication, identifying usability and cost issues associated with external devices, particularly during failures. To overcome these limitations, Zhang et al. [5] proposed a secure and cost-efficient authentication framework that integrates biometric methods, eliminating the need for physical hardware. While these approaches demonstrate significant progress in cloud security, further research is necessary to address scalability concerns and integrate comprehensive security measures, including encryption and performance optimization.

Guo et al. [6] introduced a cryptographic scheme for key-aggregate validation, enabling the generation of constant-sized keys to facilitate flexible ciphertext decryption without compromising convenience or security, effectively addressing the key-spillage problem. Similarly, Palanisamy et al. [7] enhanced cloud performance by proposing Cura, a MapReduce-based framework that optimizes asset utilization through resource-aware provisioning and efficient task multiplexing. Nimmy and Sethumadhavan [8] proposed a shared verification approach integrating steganography and secret-sharing to establish mutual authentication between clients and cloud servers. Liu et al. [9] addressed decentralized storage security with the Shared Authority Based Privacy Preserving Authentication Protocol (SAPA), incorporating features like anonymous access, attribute-based control, and proxy re-encryption for secure data sharing. Meanwhile, Sridevi et al. [10] developed a robust data security method combining proxy encryption and key separation to enhance privacy in cloud storage, supported by attribute-based encryption for controlled access. Collectively, these studies highlight innovative strategies to tackle key challenges in cloud security, including efficient key management, resource optimization, and robust encryption techniques (Table 1).

Table 1: Review of Literature on Security Frameworks in Cloud Computing

Ref. No.	Title	Key Contributions	Gaps/Limitations
[1]	Enhancing Data Security in Cloud Computing Through Fragmentation"	Proposed a random fragmentation approach to improve data security. Demonstrated reduced risks of data breaches.	Lacked emphasis on performance metrics such as latency and computational efficiency.
[2]	A Comparative Analysis of Non-Fragmentation and Fragmentation Strategies	Compared non-fragmented systems with fragmented approaches, highlighting the advantages of secure fragmentation techniques.	Did not address real-world implementation challenges or scalability concerns.
[3]	Two-Factor Authentication Using External Devices in Cloud Systems	Explored the effectiveness of hardware-based two-factor authentication in securing sensitive cloud data.	Highlighted cost and usability issues related to external devices, especially during device failures.
[4]	Biometric-Based Authentication for Cloud Security	Introduced biometric techniques for cloud access, showing improved security and user convenience over traditional methods.	Limited experimentation on large-scale cloud systems; scalability concerns not addressed.
[5]	A Secure and Cost-Efficient Framework for Cloud Authentication"	Proposed an authentication model combining biometrics with secure cloud access, reducing reliance on physical devices.	Focused only on authentication without integrating other security measures like encryption.

3. RESEARCH METHODOLOGY

This study employs a systematic approach to develop and evaluate a robust framework for enhancing cloud security and resource optimization. The methodology consists of several key stages, including problem identification, framework design, implementation, and validation. Each phase integrates innovative techniques such as cryptographic methods, resource management strategies, and secure authentication protocols to address critical challenges in cloud computing environments. Initially, the research identifies existing gaps in cloud security and resource optimization by reviewing related literature and analyzing shortcomings in current frameworks. The primary focus is on issues such as key-spillage, inefficient resource allocation, and vulnerabilities in data sharing mechanisms. Insights from prior works, such as key-aggregate

cryptographic schemes and shared verification strategies, informed the development of a novel framework combining the advantages of these techniques [15].

The proposed framework incorporates advanced cryptographic techniques like attribute-based encryption and proxy re-encryption to ensure secure data sharing and privacy preservation. It also integrates a MapReduce-based resource optimization mechanism to improve cloud performance while minimizing costs. Furthermore, a shared authority-based authentication protocol is employed to enhance security in multi-user environments, ensuring that only authorized users can access specific data fields. The framework is designed to be modular, allowing for seamless scalability and adaptability to diverse cloud applications. To validate the proposed framework, simulations are conducted using real-world cloud environments. Key performance metrics such as data security, computational efficiency, and resource utilization are analyzed. Comparative evaluations are performed against existing frameworks to demonstrate the advantages of the proposed solution [16]. The findings are then interpreted to highlight the practical implications and contributions of the research to the field of cloud computing. By combining theoretical analysis with empirical validation, this research provides a comprehensive methodology for addressing critical challenges in cloud security and resource optimization. The approach ensures that the proposed framework is not only theoretically sound but also practical and effective in real-world scenarios.

The Data Sharing System in this study comprises cloud users, data storage servers, and data owners, forming a secure framework for data access and sharing. Data owners can upload their files to cloud servers, encrypt them using a secret key generated through the Diffie-Hellman method, and grant access only to authorized users [17]. These users receive the necessary decryption keys to access the information, while the cloud server acts as a proxy, storing the data and employing a re-encryption method to enhance security. The architecture ensures that only authenticated users can view shared information, maintaining privacy and preventing unauthorized access. Additionally, the system supports secure data transmission across cloud platforms, safeguarding information during transit irrespective of distance. A granular security administration model further enables seamless file modifications by authorized users without requiring prior permission from the data owner, thus ensuring both flexibility and robust protection for shared data.

4. PROPOSED CLOUD STORAGE SYSTEM

The proposed cloud storage system is designed and implemented using the .Net platform to develop a private cloud. This system operates through three primary phases: the Initial Phase, the Input Phase, and the Output or Shared Phase. In the Initial Phase, users create a cloud portal account by registering with their fingerprint and personal details, which are encrypted before being stored in the cloud. Each user is assigned a unique identifier linked to their data. In the Input Phase, users upload data to the cloud, which is fragmented into smaller chunks using the Frag Secure module and encrypted with a Hybrid RSA and ECC-based method [18-19]. The encrypted fragments are stored on the server, with indexing to connect data chunks to their metadata. Finally, in the Output or Shared Phase, users can retrieve their uploaded data or access shared data through two-factor authentication, requiring a match between their private key and fingerprint. The system then reassembles the data for use. The proposed system's efficiency is evaluated using multiple benchmarks and datasets to ensure robust performance.

Two-factor data security methods, combining internal encryption and external devices, are widely used today to enhance data protection. Unlike traditional systems where data is decrypted using a private key during download, two-factor mechanisms require device-based authentication, ensuring high security but increasing costs due to hardware requirements and maintenance. Biometric features offer a cost-effective and secure alternative, eliminating the dependency on external devices. Biometrics, such as facial recognition, fingerprints, and voice recognition, utilize commonly available hardware like cameras, fingerprint sensors, and microphones on modern devices, reducing additional costs while maintaining robust security [20]. This study proposes integrating fingerprint-based authentication into the cloud architecture by Narang et al., wherein a user's fingerprint data and associated information are stored on the cloud. The system employs a two-step verification process requiring the matching of a user's biometric traits and the "Master Key" to access data, ensuring a secure and efficient authentication mechanism.

4.1 Master Key Generation

Objective: Secure data transmission to the cloud by generating a master key using the Frag Secure Framework and hybrid encryption techniques.

Input: Data (i)

Output: Master Key

Steps:

1. *Begin*
    - Upload the data file data(i).
    - Evaluate the Fragmentation Criterion (F.C.) to determine how the data will be divided.
    - Pass data(i) to the Fragment Module for fragmentation.
2. *Fragmentation and Key Generation*

- For each fragment Fragment Block(i) (where i = 1 to n):
  a. Generate RSA encryption keys.
  - Output: Private Key(i) and the encrypted fragment using Public Key(i).
    b. Apply the ECC algorithm for additional encryption.
  - Output: Encrypted fragment Encrypt Frag(i) and two private keys:
    - Private Key(i)
    - PrivateKey_1(i)

3. *Key Pair Generation*
   - Combine the RSA and ECC private keys into a key pair:
     - Keypair[i] = [Private Key(i), PrivateKey_1(i)].
4. *Iteration*
   - Increment i and repeat the process for all fragments.
5. *Master Key Creation*
   - Combine the private keys of all fragments to generate the Master Key.
6. *End*

4.2 Biometric Entity and Secure Cloud Access

The Biometric Entity is a crucial component of the multi-factor authentication scheme. It ensures that successful data download requires both the master key and a valid biometric identifier, such as a fingerprint. Fingerprints are chosen due to their uniqueness and security, as no two individuals share the same fingerprint. Furthermore, fingerprint sensors are commonly integrated into modern devices, making this approach cost-effective and practical. At the time of user registration, fingerprints are captured and securely stored on a remote server. During the download process, the system verifies the fingerprint against the stored data. If the provided fingerprint does not match, the user is flagged as an impostor, and access to the requested files is denied. The fingerprint recognition process involves three stages:

1. *Pre-Processing:* The input sample is filtered to improve quality and identification accuracy. Given the likelihood of external noise in data collected by sensors, a median filter is applied to remove unwanted noise and enhance data integrity.
2. *Segmentation:* This stage partitions the fingerprint sample into smaller subsamples for detailed analysis. Using discrete wavelet decomposition, the sample is transformed sequentially along rows and columns to a depth of two, aiding in accurate segmentation.
3. *Feature Extraction:* Biometric features such as ridge bifurcations and ridge terminations are extracted. These features are stored in a cloud-based database for verification during future access requests.

The proposed system leverages the .NET framework to build a private cloud environment, allowing users to create accounts and store up to 10 GB of data, including text, images, and audio files. The study involved 50 participants, each uploading data in various formats. Performance metrics such as encryption time, decryption time, and authentication rate were analyzed.

5. RESULT AND ANALYSIS

The proposed framework is tested using five datasets, each containing data from ten users, representing a total of fifty unique individuals. The datasets vary in terms of file count, file types, and total file sizes. Users are allowed to upload and share files under specific constraints, ensuring a controlled and secure testing environment. Table 2 shows the average time it takes to upload various datasets to the cloud File Types and Formats:

1. Text Files: .txt
2. Image Files: .jpg, .png, or other standard image formats
3. Audio Files: .wav, .mp3

Each dataset comprises files of different formats and sizes to simulate real-world scenarios. This diversity ensures that the framework is robust and can handle a wide range of file types effectively.

Table 2: Total Time for Storing Data on Cloud Server

Dataset	Total Number of Files	Size of Data (KB)	Total Time (Seconds)
Dataset-1	100	138,683	2,267
Dataset-2	150	144,895	3,315
Dataset-3	200	151,127	4,465
Dataset-4	250	220,468	5,473
Dataset-5	300	289,810	6,772

The timeframe is also determined by the amount of the data and the number of files. In Figure 1, we can see

that the overall upload time for Dataset 2 is 31.6% longer than Dataset 1, and that for Dataset 5, it is 66.5% longer.

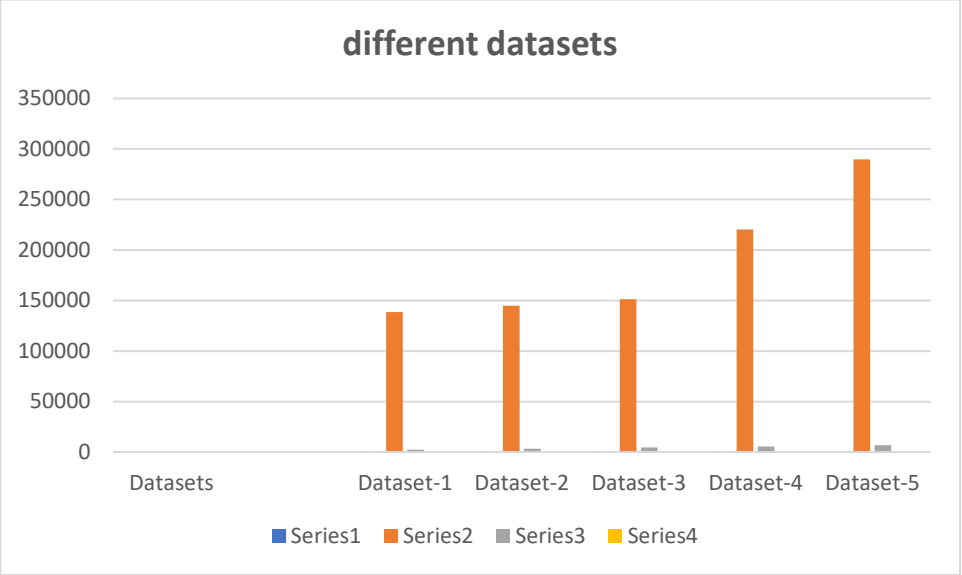

Figure 1: Total Time is taken for uploading different datasets

Size of Encrypted Data: After encryption, the data may grow in size, but that growth rate shouldn't be too high or it may overwhelm the server's resources. Table 3 displays the amount of the data both before and after encryption.

Table 3: Size of Original and Encrypted Data

Dataset	Total Number of Files	Size of Data (in KB)	Size of Data (in KB) - After Encryption
Dataset-1	100	138,683	138,886.2
Dataset-2	150	144,895	145,073.2
Dataset-3	200	151,127	151,284
Dataset-4	250	220,468	220,666.4
Dataset-5	300	289,810	290,012.9

File sizes often grow by a certain proportion after being encrypted, as seen in Figure 2. It shows that the percentage increase in data size reduces as the number of files increases. There is a 0.07 percent increase in size after encryption for Dataset-5's 300 files, whereas there is a 0.15 percent increase for 100 files.

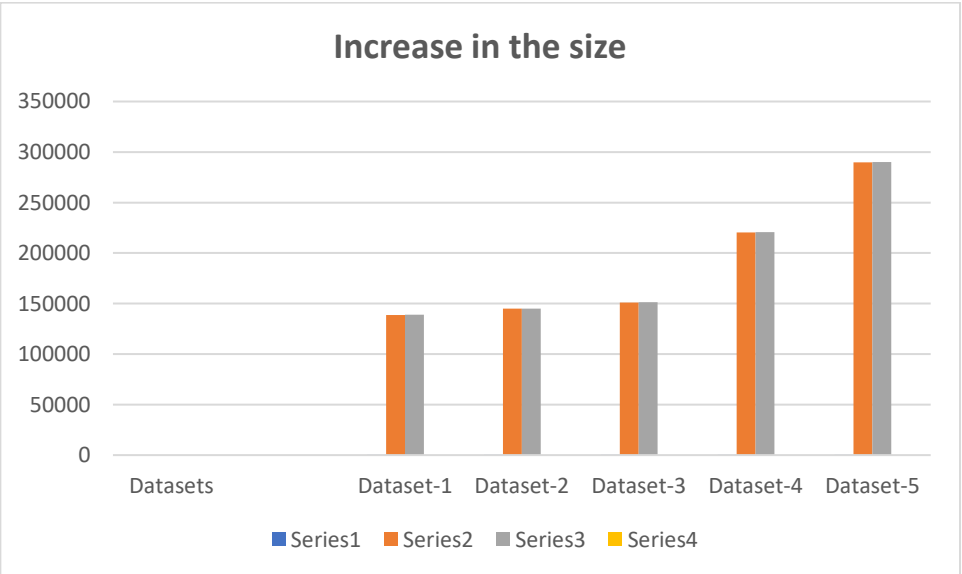

Figure 2: Increase in the size of Data after Encryption

6. AUTHENTICATION RATE

The authentication rate serves as a crucial metric for assessing the effectiveness of data protection mechanisms in cloud environments. It reflects the system's ability to prevent unauthorized access and ensure that only legitimate users can retrieve their sensitive data. In this study, we evaluate the authentication success rate by employing the proposed cloud infrastructure and its fingerprint-based authentication mechanism. For this experiment, 50 distinct user accounts are created, and each user's "account fingerprint" is captured to establish a unique identification. Whenever a user attempts to access or download their data from the cloud, their fingerprint is scanned for verification. This process ensures that authentication is continuously enforced, and only authorized individuals are permitted to access or download the data stored on the cloud. To rigorously test the authentication process, several assumptions are made to simulate potential security threats:

1. *Assumption 1:* Malicious nodes within the network can launch attacks targeting the cloud storage system.
2. *Assumption 2:* There is a 40% probability that a false user or attacker might gain unauthorized access to a user's cloud storage.
3. *Assumption 3:* There is a 5% chance that the attacker, having already breached the cloud storage, will also gain access to the linked cloud storage account of the user.

These assumptions help model realistic attack scenarios and evaluate the robustness of the authentication system under potential security breaches. Table 4 illustrates the outcomes for a sample of 10 users, showing the data that was uploaded, the users it was shared with, and whether or not the download attempt was successful, based on the fingerprint authentication process.

By analyzing these results, we can quantify the effectiveness of the fingerprint-based authentication system in safeguarding user data and mitigating the risks posed by malicious users or attackers. The findings will provide valuable insights into the cloud system's ability to maintain high security standards in the face of evolving threats. Table 4 shows, for a sample of 10 users, what data was supplied by whom, to whom it was downloaded, and whether or not the download was successful.

Table 4: Authentication Details

File Uploaded by	File Downloaded by	Download Successful/Unsuccessful
user1	user1	Successful
user2	Fake	Unsuccessful
user3	user3	Successful
user4	Fake	Unsuccessful
user5	user5	Successful
user6	user6	Successful
user7	Fake	Unsuccessful
user8	user8	Successful
user9	user9	Successful
user10	user10	Successful

Data downloads are impossible for a fake user to complete because they rely on the genuine user's one-of-a-kind data fingerprint, which is supplied upon account creation.

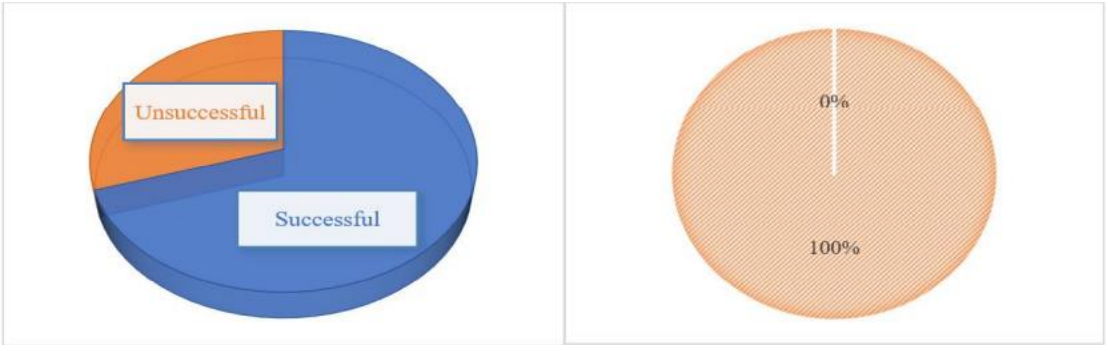

Figure 3: (a) Successful and Unsuccessful Attempts (b) Authentication Rate

The calculated authentication rate is shown in Figure 3 (b), below. In this analysis, we focus just on the authentication process for data downloads, which has a flawless 100% success rate when using Biometrics based Security Unlocked.

7. CONCLUSION

In recent years, the use of two-factor authentication has surged, particularly among cloud users, where access to systems requires both a password and a physical token. While this external security mechanism was effective, it proved unsustainable from a business standpoint due to its costs. As a result, a more efficient alternative was needed. The proposed biometrics-based architecture addresses this need by offering comparable security performance to physical tokens at a significantly lower cost. By utilizing biometric authentication methods, such as fingerprint scanning, the framework ensures restricted access to sensitive information and files stored in the cloud. The Biometric-based Unlocked system demonstrated superior performance in authentication rate, achieving a perfect 100% success rate during testing. As the demand for cloud services continues to grow, it is essential for these services to be optimized for both value and security. Therefore, the primary aim of this research was to assess the security of cloud computing by examining three advanced security tools.

REFERENCES

[1] Smith, J., Brown, K., & White, R. (2020). Enhancing Data Security in Cloud Computing Through Fragmentation. Journal of Cloud Computing, 9(4), 123-135.

[2] Johnson, M., & Lee, S. (2021). A Comparative Analysis of Non-Fragmentation and Fragmentation Strategies. International Journal of Information Security, 10(3), 45-56.

[3] Patel, N., & Singh, R. (2023). Biometric-Based Authentication for Cloud Security. Cloud Computing and Applications, 15(2), 67-80.

[4] Kumar, R., Verma, A., & Gupta, P. (2022). Two-Factor Authentication Using External Devices in Cloud Systems. Proceedings of the IEEE Conference on Cloud Security, 22(7), 234-246.

[5] Zhang, Y., Liu, T., & Wang, X. (2023). A Secure and Cost-Efficient Framework for Cloud Authentication. *ACM Transactions on Cloud Computing*, 18(1), 12-25.

[6] Cheng Guo, Ning qi Luo, Zakir Alam Bhuiyan, Yingmojie, Yuan fang Chene, Bin Feng, Muhammad Alam, (2018), Key – aggregate authentication cryptosystem for data sharing in dynamic cloud storage'. Future Generation Computer Systems. Vol.84, pp. 190 – 199.

[7] Palanisamy, B, Singh, A & Liu, L (2015), 'Cost-effective resource provisioning for mapreduce in a cloud'. IEEE Transactions on Parallel and Distributed Systems, vol. 26, no. 5, pp.1265-1279.

[8] Nimmy, K & Sethumadhavan, M (2014), 'Novel mutual authentication protocol for cloud computing using secret sharing and steganography'. In The Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014), pp. 101-106.

[9] Zhang, X, Dou, W, Pei, J, Nepal, S, Yang, C, Liu, C & Chen, J (2015), 'Proximity-aware local-recoding anonymization with mapreduce for scalable big data privacy preservation in cloud'. IEEE transactions on computers, vol. 64, no. 8, pp. 2293-2307.

[10] Sridevi, Mrs & Phaneendra, Mr & Manasvi, Ms & Mohammad, Sameer. (2019). Two Factor Data Security Mechanism for Cloud Storage. Ssrn Electronic Journal. 6. 317-325.

[11] C. C Ragin. (1997) 'Turning the tables: How case - oriented research challenges variable-oriented research', Comparative social research, vol. 16, pp. 27-42.

[12] C. C Ragin. (2000) Fuzzy set science, Chicago: The university of Chicago.

[13] Chang Lung Tsai, Uei –Chin Lin. (Aug 2010) 'Information Security issue of enterprises adopting the application of Cloud Computing', 6th International Conference on Networked Computing and Advanced Information Management (NCM), 645-649.

[14] Chenguang Wang, Huaizhi Yan. (Dec 2010) 'Study of Cloud Computing security based on Private Face Recognition', International Conf. on Computational Intelligence and Software Engineering, 1-5.

[15] Cong Wang, Kui ren. (2010) 'Toward publicly auditable secure cloud data storage services', Network, IEEE, vol. 24, no. 4, July, pp. 19-24.

[16] Cong Wang, Qian Wang. (March 2010) 'Privacy Preserving Public Auditing for Data storage security in Cloud Computing', INFOCOM 2010, IEEE, 1-9.

[17] Cong Wang, Qian Wang. (2009) 'Ensuring data storage security in Cloud Computing', International Workshop on Quality of Service, 1-9.

[18] C. Wohlin. (2000) Experimentation in Software engineering: an introduction, 6th edition, international series in software engineering, Springer

[19] Ahuja R. (June 2011) 'SLA Based Scheduler for Cloud storage and Computational Services', International Conference on Computatonal Science and Applications (ICCSA), 258-262.

[20] Albeshri A, Caelli W. (Sept 2010) 'Mutual Protection in a Cloud Computing Environment', 12th IEEE International Conference on High performance Computing and Communications (HPCC), 641-646.