Cloud based Image Processing Services Using Secure Multiparty Computation

Mohammed Awais Ahmed[1]
[1] *Student,* BE [IT] Dept of Information Technology, *LIET Hyderabad TS, India*

***Abstract*—Searchable Encryption (SE) schemes have gained significant attention in providing security and privacy for cloud data. Existing SE approaches enable multiple users to perform search operations using schemes like Broadcast Encryption and Attribute-Based Encryption (ABE). However, these schemes lack support for multiple users to search over the encrypted data of multiple owners. Moreover, approaches involving a Proxy Server (PS) impose a heavy computational burden due to repeated query encryption for transformation purposes.**

To address these limitations, this project proposes a novel secure proxy server approach that enables search operations without query transformation, alleviating the computational burden on the PS. The proposed approach employs the Euclidean distance similarity method to provide users with the top-k relevant documents for their queries.

Extensive experiments were conducted to evaluate the efficiency of the proposed approach in terms of search time and accuracy. The results demonstrate that the secure proxy server approach outperforms existing methods, achieving faster search times and improved accuracy. The elimination of query transformation contributes to the reduction of computational overhead on the PS, leading to enhanced scalability and user experience in multi-user environments

I. INTRODUCTION

In today's data-driven world, the field of data science has gained immense popularity. With the massive volume of data generated in various sectors such as retail, healthcare, and education, there is a growing need to perform operations and extract meaningful insights from these datasets. Data science encompasses a multidisciplinary approach that involves mathematical modeling, statistical analysis, data visualization, and the use of advanced technologies to handle both structured and unstructured data.

Java, known for its versatility and robustness, has emerged as a powerful programming language for data science tasks. Its extensive collection of libraries and built-in features make it well-suited to address the diverse challenges faced in the field. In this paper, we explore the data science process and delve into how Java can effectively tackle the demands of data analysis, modeling, and visualization. We discuss the various scenarios where data science is applied, such as recommendation systems, financial risk management, healthcare services improvement, computer vision, and efficient energy management. Furthermore, we highlight the key features of Java that make it a preferred choice for data science projects, including its simplicity, cross-platform compatibility, execution speed, memory management capabilities, and the availability of specialized libraries.

II. LITERATURE SURVEY

1. A KNN query processing algorithm using a tree index structure on the encrypted database by Hyeong-II kim, Hyeong- JIn kim, Jae-Woo Change

Database outsourcing in cloud computing raises significant privacy concerns, necessitating encryption of the data before outsourcing. As a result, several KNN query processing techniques have been proposed for encrypted databases. This paper focuses on addressing these concerns by proposing improved KNN query processing algorithms for encrypted databases. The proposed algorithms ensure the confidentiality of the encrypted data and user query records while achieving high query processing efficiency

2.Efficient query processing on outsourced encrypted dat ain cloud with privact preservation by B.R.Purushothama, B.B.Amberker. This work addresses security challenges in data outsourcing to public clouds and focuses on maintaining confidentiality and privacy of sensitive data. Two efficient solutions are proposed for query processing on encrypted data, prioritizing query performance and data privacy. Adversaries are limited to minimal information, and empirical evaluations demonstrate

the superior efficiency of the proposed schemes compared to existing approaches.

3 Hilbert-curve based cryptographic transformation scheme for protecting data privacy on outsourced private spatial data by Hyeong-Il Kim, Seung-Tae Hong, Jae-Woo Chang. This paper focuses on protecting location data privacy in outsourced databases for cloud computing. Existing spatial transformation schemes are vulnerable, and cryptographic schemes have high query processing costs. The proposed Hilbert-curve based cryptographic transformation scheme enhances data privacy and improves query processing efficiency through local clustering. Performance analysis demonstrates its superiority over existing schemes.

4 Controlling outsourcing data in cloud computing with attribute-based encryption by Shuaishuai Zhu, Yiliang Han, Yuechuan Wei. Cloud computing has gained prominence in the IT society, leading to increased data outsourcing. However, data privacy and control are major concerns for cloud users, especially large companies. This paper introduces a tree-based dataset management model to address storage, sharing, and ownership issues in cloud computing. The model incorporates strategies such as data encryption, boundary maintenance, and data proof to ensure entity privacy, data availability, and secure data sharing. It offers a flexible data management mechanism within the cloud environment

III. PROPOSED SYSTEM

A cloud server is assigned the task of storing all the documents and indices from different owners and when a search request from a data user is received, it needs to find the most relevant documents and return them to the data user. A data owner creates an index for each of its documents. It encrypts the document collection and sends the encrypted documents over to the cloud server. The words in the indices are partially encrypted with the owner's secret key and then these indices are sent to the proxy server. A proxy server is given the work of completing the encryption of partially encrypted index words as well as query keywords before they are sent to the cloud server. The proxy server has a key, known to only it, that is used as a common key to complete the encryption of all the partially encrypted words received. A data user's task is to frame search queries and to partially encrypt these query keywords with its own secret key before sending them to the proxy server.

Fig 1.1 Software development cycle

IV. HARDWARE AND SOFTWARE REQUIREMENTS

A. HARDWARE REQUIREMENTS

- System : Pentium IV 2.4 GHz.
- Hard Disk : 500 GB
- Monitor : 15 VGA Color.
- Mouse : Logitech.
- Ram : 1 GB.

B. SOFTWARE REQUIREMENTS

- Operating system : Windows /7.
- Coding Language : JAVA/J2EE
- IDE : NetBeans 7.2
- Database : MYSQL

V. SYSTEM DESIGN

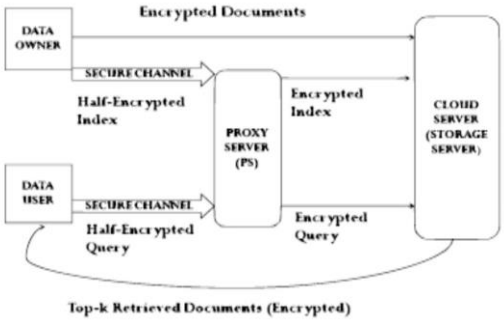

Fig 1.2 System architecture

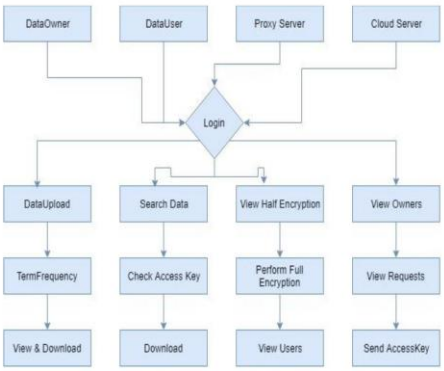

Fig 1.3 Data flow diagram

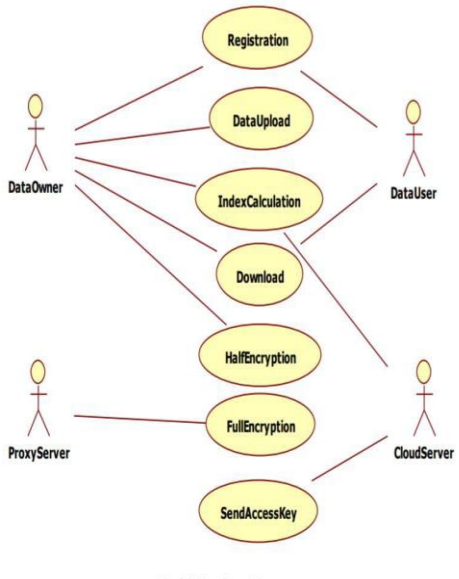

Fig 1.4 Use case diagram

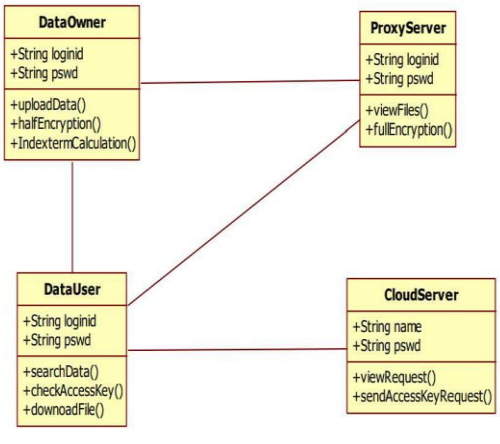

Fig 1.5 Class diagram

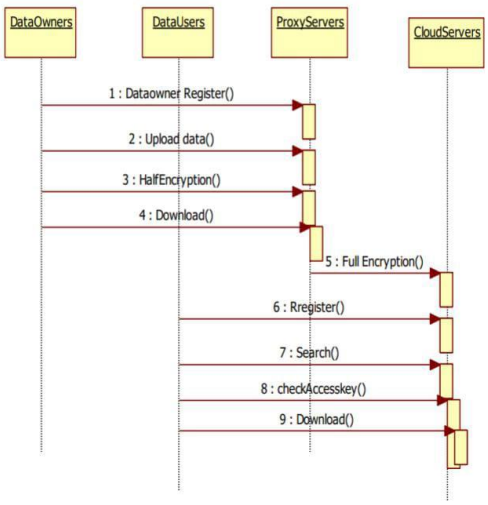

Fig 1.6 Sequence diagram

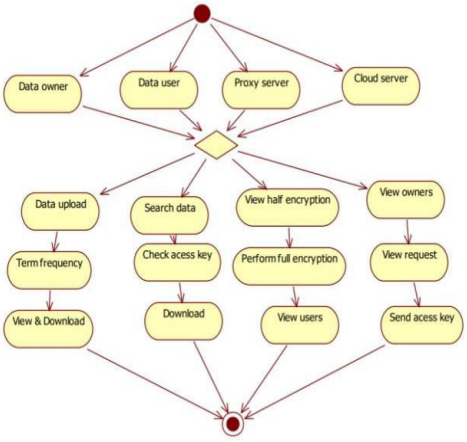

Fig 1.7 Activity diagram

VI. RESULT

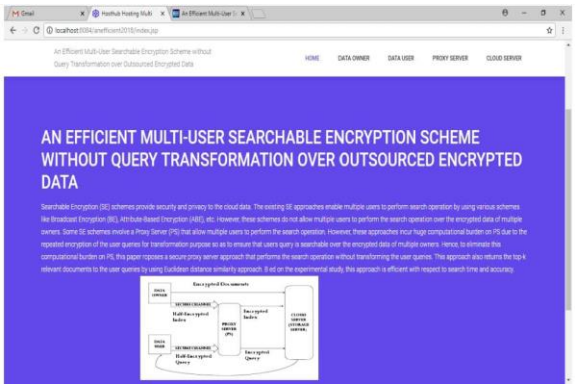

Fig 1.8 Homepage

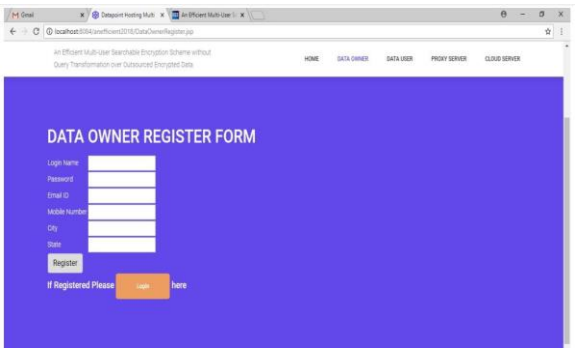

Fig 1.9 Data owner register form

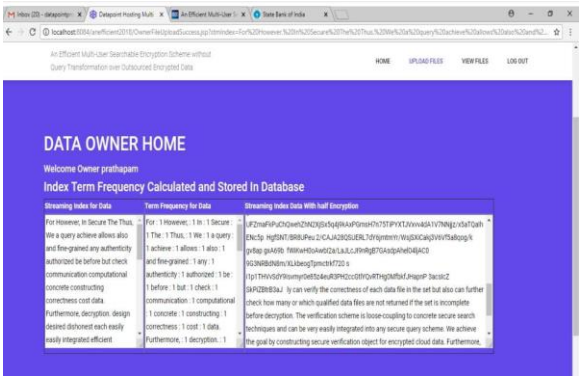

Fig 1.10 Data owner home

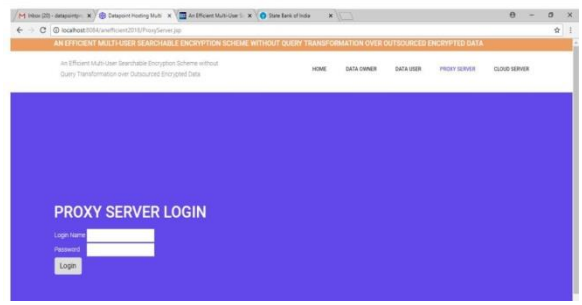

Fig 1.11 Proxy server login

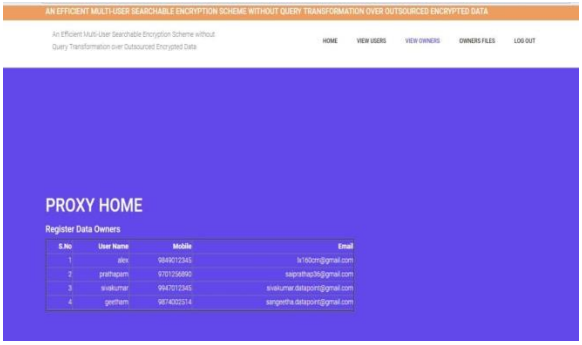

Fig 1.12 View registered data users

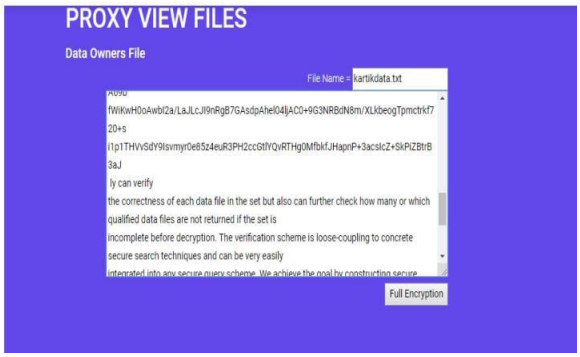

Fig 1.13 Data user file half encryption view

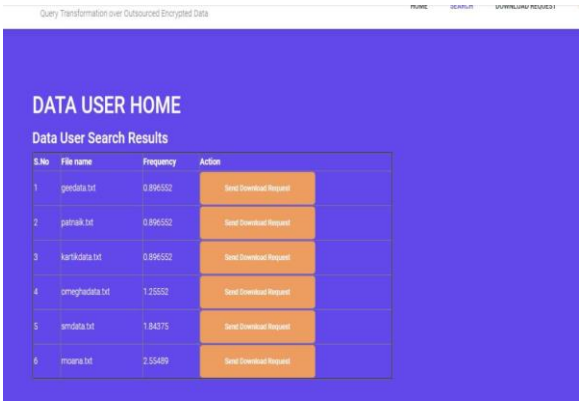

Fig 1.14 Data user home

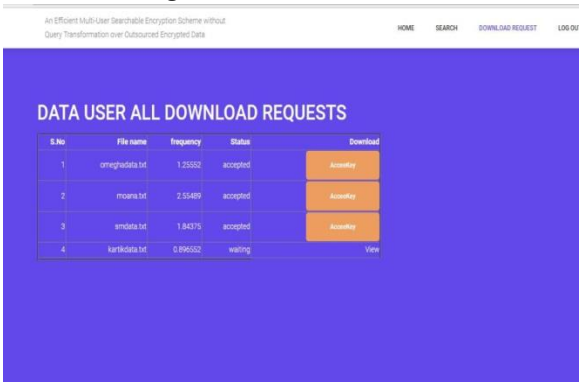

Fig 1.15 Download request sent

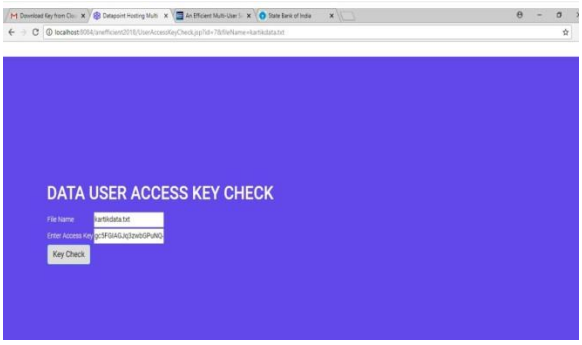

Fig 1.16 Downloading file

VII. CONCLUSION

This paper proposes a Proxy server-based approach for enabling search operations over the data of multiple owners. Unlike existing approaches, this approach allows the data user's query to search across multiple owners' data without query transformation. The approach utilizes partial encryption, where half of each index keyword and query keyword is encrypted using the respective secret keys, while the other half is encrypted using a common secret key of the proxy server. Experimental results validate the efficiency of the proposed approach. Future work may include adding a module for data user addition and revocation, as well as enhancing the security functionalities of the approach.

REFERNCES

[1] D. X. Song, D. Wagner, and A. Perrig, "Practical techniques for searches on encrypted data," in Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on. IEEE, 2000, pp. 44–55.

[2] D. Boneh, G. Di Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search," in International Conference on the Theory and Applications of Cryptographic Techniques. Springer, 2004, pp. 506–522.

[3] J. Lotspiech, "12 - broadcast encryption," in Multimedia Security Technologies for Digital Rights Management, W. Zeng, H. Yu, and C.-Y. Lin, Eds. Burlington: Academic Press, 2006, pp. 303 – 322.

[4] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proceedings of the 13th ACM Conference on Computer and Communications Security, ser. CCS '06. New York, NY, USA: ACM, 2006, pp. 89–98.

[5] W. Zhang, Y. Lin, S. Xiao, J. Wu, and S. Zhou, "Privacy preserving ranked multi-keyword search for multiple data owners in cloud computing," IEEE Transactions on Computers, vol. 65, no. 5, pp.1566–1577, 2016.

[6] R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, "Searchable symmetric encryption: improved definitions and efficient constructions," CCS-2006:ACM conference on Computers and Communications Security, pp. 79–88, 2006.

[7] Q. Wang, Y. Zhu, and X. Luo, "Multi-user searchable encryption with fine-grained access control without key sharing," in 2014 3rd International Conference on Advanced Computer Science Applications and Technologies, Dec 2014, pp. 145–150.

[8] Z. Deng, K. Li, K. Li, and J. Zhou, "A multi-user searchable encryption scheme with keyword authorization in a cloud storage," Future Generation Computer Systems, vol. 72, pp. 208–218, 2017.

[9] T. Korenius, J. Laurikkala, and M. Juhola, "On principal component analysis, cosine and Euclidean measures in information retrieval," Information Sciences, vol. 177, no. 22, pp. 4893 – 4905, 2007.

[10] RFC, "Request for comments database," https://www.rfceditor. org/retrieve/bulk/.