A PRIVACY-FOCUSED FRAMEWORK FOR SECURE SEARCHABLE ENCRYPTION IN CLOUD STORAGE

Bairy Kumaraswamy
Scholar, Department of MCA
Vaageswari College Of Engineering-Karimnagar

P. Sathish
Assistant Professor, Department of MCA
Vaageswari College of Engineering-Karimnagar

Dr.V. Bapuji
Professor & Head, Department of MCA
Vaageswari College of Engineering-Karimnagar

ABSTRACT: Searchable encryption has received a significant attention from the research community with various constructions being proposed, each achieving asymptotically optimal complexity for specific metrics (e.g., search, update). Despite their elegance, the recent attacks and deployment efforts have shown that the optimal asymptotic complexity might not always imply practical performance, especially if the application demands a high privacy. In this article, I introduced a novel Dynamic Searchable Symmetric Encryption (DSSE) framework called Incidence Matrix (IM)-DSSE, which achieves a high level of privacy, efficient search/update, and low client storage with actual deployments on real cloud settings. I harness an incidence matrix along with two hash tables to create an encrypted index, on which both search and update operations can be performed effectively with minimal information leakage. This simple set of data structures surprisingly offers a high level of DSSE security while achieving practical performance. Specifically, IM-DSSE achieves forward-privacy, backward-privacy and size-obliviousness simultaneously. I also create several DSSE variants, each offering different trade-offs that are suitable for different cloud applications and infrastructures. I fully implemented our framework and evaluated its performance on a real cloud system (Amazon EC2). I have released IM-DSSE as an open-source library for wide development and adaptation.

Index Terms: **Privacy-enhancing technologies, Private cloud services, Dynamic searchable symmetric encryption.**

I.INTRODUCTION

The expansion of cloud computing and storage services is going to be very beneficial for society and the IT industry alike. To help clients with limited resources, such as individuals or small/medium enterprises, Storage-as-a-Service (SaaS) may significantly reduce the cost of data management by providing ongoing support, expertise, and maintenance.

This makes it a vital cloud service. While there are benefits to using SaaS, there are also major concerns around customer privacy and security. In other words, once a client uploads their data to the cloud, there's a greater chance that an adversary may use malware or other malicious software to access and exploit sensitive information, including emails. While standard encryption algorithms like Advanced Encryption Standard (AES) do a good job of keeping data private, they also make it impossible for clients to retrieve their encrypted data stored in the cloud. Cloud computing's benefits and utility can be drastically diminished as a result of the trade-off between data usage and privacy. As a result, I need to find a way to fix this problem and make better use of the cloud service without sacrificing privacy.

In order to facilitate keyword searches, clients have the option to encrypt data using Searchable Symmetric Encryption (SSE) [1]. It is possible to conduct these encrypted searches using "search tokens" [2] that link keywords to encrypted files in an encrypted index. SSE is often used to provide privacy-preserving keyword searches on cloud services (such as Amazon S3). Without revealing the contents of the files or queries.

a data owner might outsource the search for keywords on encrypted files [3]. Due to their inability to update and provide search-only capabilities on static data, early SSE systems (e.g., [1], [4]) have very limited use. Since then, many Dynamic SSE (DSSE) approaches have been proposed, including [3], [5], which enable the user to add or delete files once the system is set up. I am unaware of any DSSE scheme that outperforms the others in the aforementioned areas: storage efficiency, functionality, privacy (e.g., information leakage), and performance (e.g., search, update delay). Our study aims to address some of the flaws in the current state of DSSE research.

II.LITERATURE SURVEY

R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky introduced Searchable Symmetric Encryption (SSE), which allows users to orutsource their data storage while maintaining the ability to search through the data securely. They reviewed existing security definitions, identified their shortcomings, and proposed two stronger, equivalent definitions. Their new constructions not only meet these stronger security standards but also demonstrate greater efficiency than previous methods. Furthermore, they extended SSE to a multi-user setting, enabling parties other than the data owner to submit search queries, and provided an efficient construction for this scenario.

Emil Stefanov and Charalampos Papamanthou tackled Dynamic Searchable Symmetric Encryption (DSSE), which supports searchable and updatable encrypted document collections. Prior DSSE schemes either leaked significant information or were inefficient. They proposed the first DSSE scheme that balances minimal information leakage with efficiency, supporting both updates and searches in sublinear time and maintaining a linear-sized data structure. Their implementation confirmed the practical efficiency of their approach.

N. Cao, C. Wang, M. Li, K. Ren, and W. Lou focused on the challenges of enabling

privacy-preserving multi-keyword ranked search over encrypted cloud data (MRSE). They addressed the need for multi-keyword searches and relevance-ranked results. They used "coordinate matching" and "inner product similarity" to measure and evaluate document relevance. They proposed a basic MRSE scheme and two improved versions to meet strict privacy requirements under different threat models, extending the schemes to support more search semantics. Their experiments demonstrated low computational and communication overhead.

W. Sun, B. Wang, N. Cao, M. Li, W. Lou, Y. T. Hou, and H. Li presented a verifiable privacy-preserving multi-keyword text search (MTS) scheme with similarity-based ranking. To support efficient multi-keyword search and result ranking, they used term frequency, the vector space model, and cosine similarity. They improved search efficiency with a tree-based index structure and multi-dimensional algorithms, ensuring better performance than linear search. They also enhanced search privacy with secure index schemes suitable for strong threat models, and included authenticity checks for returned search results. Their extensive experiments validated the effectiveness and efficiency of their schemes.

F. Hahn and F. Kerschbaum addressed issues in dynamic searchable encryption, which should be efficient, dynamic, and highly secure. Previous schemes had drawbacks like reduced security under updates, client storage requirements, or large index sizes. They proposed the first scheme with optimal search time, minimal index size, and no need for client storage (except for the key), leaking no more information than the access pattern. Their system implementation proved highly efficient for cloud storage applications.

III.PROBLEM STATEMENT

The existing DSSE systems, while innovative, face several limitations that hinder their practical deployment and effectiveness. Firstly, no single DSSE scheme currently excels in all crucial metrics, including privacy (e.g., information leakage), performance (e.g., search and update delay), storage efficiency, and functionality. Additionally, many existing systems focus predominantly on theoretical asymptotic analyses, with some offering only prototype implementations. This theoretical emphasis restricts the understanding of their practical performance and usability. Furthermore, the lack of extensive experimental performance evaluations on real platforms makes it challenging to assess the real-world application of proposed DSSE schemes, often overlooking critical issues such as security vulnerabilities, hidden computation costs, multi-round communication delays, and storage blowup. Moreover, most efficient DSSE schemes are vulnerable to file-injection attacks, which can be easily executed by a semi-honest adversary, particularly in personal email scenarios. Lastly, forward-secure DSSE schemes with optimal asymptotic complexity often suffer from high delays due to public-key operations or significant storage blow-up on both the client and server sides, casting doubt on their ability to meet the practical needs of real systems. Addressing these

issues is crucial for developing a DSSE framework that is both theoretically sound and practically viable, ensuring high security, efficient performance, and compatibility with existing cloud storage infrastructures.

IV.METHODOLOGY

In this project, we address the disparity between theoretical advancements and practical implementations in DSSE research by introducing IM-DSSE, a fully-implemented Incidence Matrix-based DSSE framework. IM-DSSE is engineered to cater to the demands of real-world privacy-critical cloud systems, emphasizing high security against practical attacks and low end-to-end delay. Our framework features a meticulously designed incidence matrix-based data structure, augmented with two hash tables, facilitating efficient and secure search and update operations. Notably, our implementation prioritizes parallel processing, enhancing the performance of these operations. We offer a range of DSSE schemes within the IM-DSSE framework, including a preliminary scheme and extended versions tailored to diverse application requirements and cloud data storage infrastructures. The advantages of our proposed system are manifold: it boasts enhanced security against File-Injection Attacks, updates with improved features, and full parallelizability. To validate its efficacy, we conduct detailed experimental evaluations on real cloud platforms like Amazon EC2, assessing metrics such as search and update latency, storage efficiency, and resistance to attacks. Furthermore, we make our framework available as open-source, promoting transparency and enabling further research and development in the DSSE domain. Through these efforts, we aim to offer a DSSE framework that seamlessly integrates theoretical rigor with practical applicability, ensuring robust security, efficient performance, and compatibility with contemporary cloud storage architectures.

V.SYSTEM ARCHITECTURE

Considering the recent incidents of data breaches (e.g., Ashley Madison, Apple iCloud, and Equifax), it is imperative to ensure data secrecy in the cloud. A number of DSSE systems have been developed; however, as noted in §1, they are not suitable for implementation on current cloud storage architectures due to their vulnerability to real-world attacks. Moreover, the majority of earlier methods are incompatible with storage-only cloud services like Dropbox and Google Drive, and deployment may be costly when using specialized computing resources. In this research, I provided a unique DSSE architecture that can achieve high security while being compatible with current cloud infrastructure. Among the many potential applications of our architecture is the provision of private email and file storage services. With these services, customers may transmit, receive, read, and modify private data on the cloud, including emails, pictures, and financial transactions, all without the provider's knowledge. Figure 1 depicts our IM-DSSE architecture from a high-level viewpoint for data storage applications.

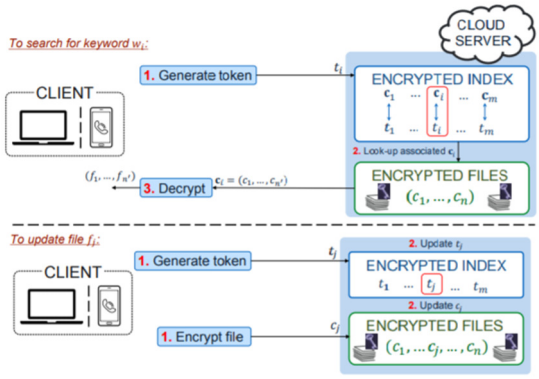


Fig. 1: IM-DSSE framework for file-storage services.

VI. MODULE DESCRIPTION

1.Data User

The user is one of the module, here the user should register with the application and should authorized by the cloud then only the user can able to search for the file, if you find the file then you should get the decryption key to view the file.

To get the decryption key, the user should request for that key to the cloud. after getting the decrypt key from the cloud the user can view the decrypted file and if the user wants to download the file here also the user should get the file token from the owner, after verifying file token the user can able to download the file.

2.Data Owner

Here the data owner is the module, the data owner should register with our application and the owner can perform the following action such as upload the file into the clouds, view uploaded files and the owner can able to check his transaction and the owner give the token to the user.

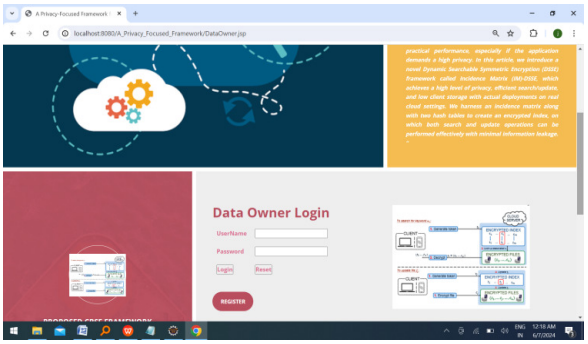
3.Cloud

The cloud is module who manage everything about the project like authorizing the users and authoring the owner, view decrypt key request, view uploaded files, view secure data details, view file attackers and view transactions.

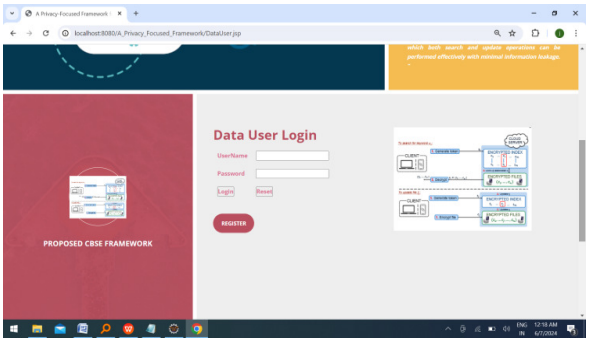
VII. RESULTS

Index Page

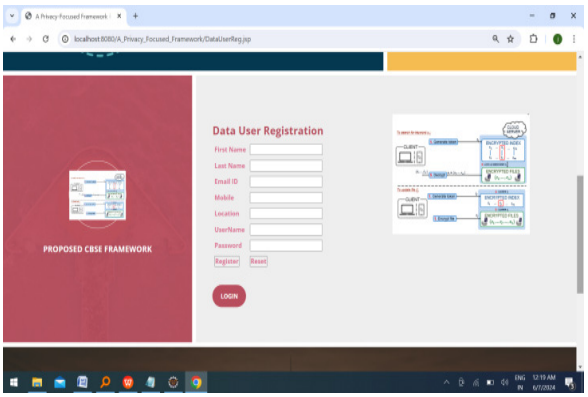
Data Owner Login



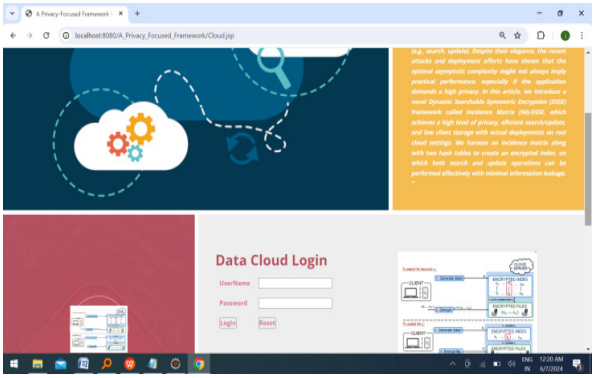
Data User Login Page



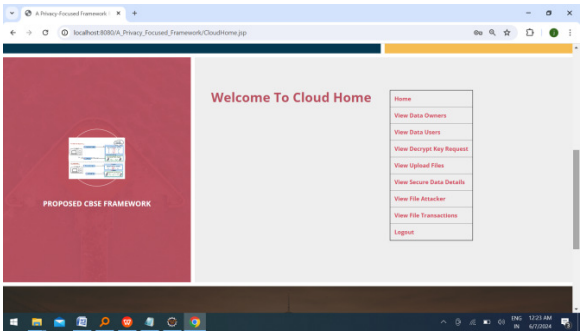
Data User Registration



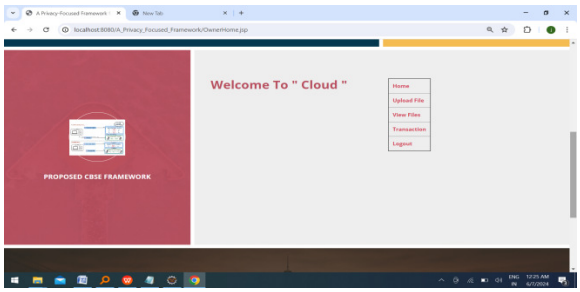
Cloud Login Page



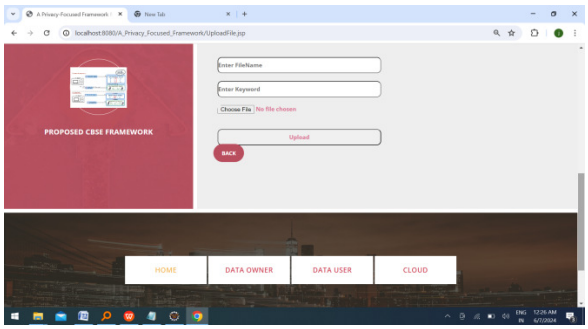
Cloud Home Page



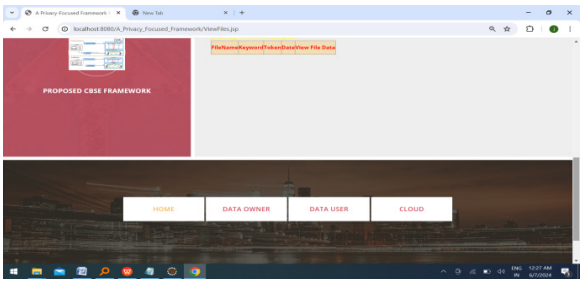
Data Owner Home Page



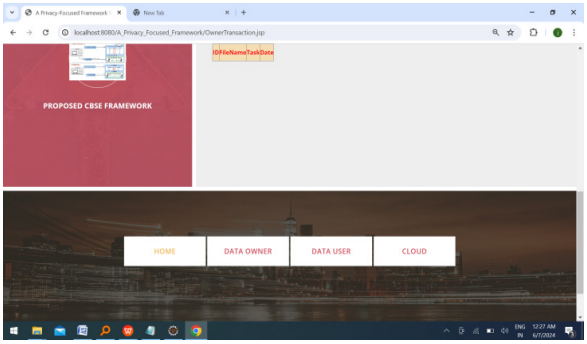
Upload File Page



View File Page



View Transaction Page



VIII.CONCLUSION

This is introduced IM-DSSE, a novel DSSE framework, in this paper. It provides low search latency, efficient updates, and excellent privacy all at once. To enable fast and safe search and update operations in our constructs by combining two hash tables with a simple yet efficient incidence matrix data structure. To accommodate cloud infrastructure and individual use in a wide range of contexts and applications, the IM-DSSE framework offers a number of DSSE architectures. When compared to competing frameworks, all of the schemes that make up the IM-DSSE architecture provide the highest levels of privacy and security. This is demonstrated our framework's great usability, especially when used on mobile devices with large datasets, through extensive experimental investigation on actual Amazon EC2 cloud systems. A fully functional version of our system is now available for everyone to use and study.

IX.FUTURE ENHANCEMENT

Future research can optimize IM-DSSE for specific data types and queries, integrate machine learning for enhanced search efficiency, and improve security against advanced attacks. Expanding support for complex queries and integrating with other privacy-preserving technologies can further broaden its applicability and utility in diverse cloud computing scenarios.

X.REFERENCES

1. R. Curtmola, J. Garay, S. Kamara, and R. Ostrovsky, “Searchable symmetric encryption: improved definitions and efficient constructions,” in Proc. 13th ACM Conf. Comput. Commun. security, ser. CCS ’06. ACM, 2006, pp. 79–88.
2. E. Stefanov, C. Papamanthou, and E. Shi, “Practical dynamic searchable encryption with small leakage,” in 21st Annu. Network and Distributed System Security Symp. — NDSS 2014. The Internet Soc., February 23-26, 2014
3. S. Kamara, C. Papamanthou, and T. Roeder, “Dynamic searchable symmetric encryption,” in Proc. 2012 ACM Conf. Comput. Commun. security. New York, NY, USA: ACM, 2012, pp. 965–976.
4. D. X. Song, D. Wagner, and A. Perrig, “Practical techniques for searches on encrypted data,” in Proc. 2000 IEEE Symp. Security and Privacy, 2000, pp. 44–55.
5. Sathish Polu and Dr. V. Bapuji, “Distributed Denial of Service (DDOS) Attack Detection in Cloud Environments Using Machine Learning Algorithms”, International Journal of Innovative Research in Technology, (IJIRT), Volume 9, Issue7, ISSN:2349-6002.December 2022, (UGC CARE LIST – I).
6. D. Cash, J. Jaeger, S. Jarecki, C. Jutla, H. Krawcyk, M.-C. Rosu, and M. Steiner, “Dynamic searchable encryption in very-large databases: Data structures and implementation,” in 21th Annu. Network Distributed System Security Symp. — NDSS 2014. The Internet Soc., February 23-26, 2014.
7. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, “Privacy-preserving multi-keyword ranked search over encrypted cloud data,” IEEE Trans. Parallel Distributed Syst., vol. 25, no. 1, pp. 222–233, 2014.

8. W. Sun, B. Wang, N. Cao, M. Li, W. Lou, Y. T. Hou, and H. Li, "Verifiable privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking," IEEE Trans. Parallel Distributed Syst., vol. 25, no. 11, pp. 3025–3035, 2014.
9. Bapuji V, Naik R Lakshman, Prasad M Rajendra ''cloud computing: research issues and implications'' IAES Institute of Advanced Engineering and Science, vol 2, issue 2, International Journal of Cloud Computing and Services …, 2013. https://scholar.google.co.in/citations?view_op=view_citation&hl=en&user=CC4GukQAAAAJ&citation_for_view=CC4GukQAAAAJ:RYcK_YlVTxYC
10. S. Kamara and C. Papamanthou, "Parallel and dynamic searchable symmetric encryption," in Financial Cryptography and Data Security (FC), ser. Lecture Notes in Comput. Sci. Springer Berlin Heidelberg, 2013, vol. 7859, pp. 258–274.
11. M. Naveed, M. Prabhakaran, and C. A. Gunter, "Dynamic searchable encryption via blind storage," in 35th IEEE Symp. Security Privacy, May 2014, pp. 48–62.
12. F. Hahn and F. Kerschbaum, "Searchable encryption with secure and efficient updates," in Proc. 2014 ACM SIGSAC Conf. Comput. and Commun. Security. ACM, 2014, pp. 310–320.
13. R. Bost, "Sophos – forward secure searchable encryption," in Proc. 2016 ACM Conf. Comput. Commun. Security. ACM, 2016.
14. Sathish Polu and Dr. V. Bapuji," "Mitigating Ddos Attacks in Cloud Computing Using Machine Learning Algorithms", The Brazilian Journal of Development ISSN 2525-8761, published by Brazilian Journals and Publishing LTDA. (CNPJ 32.432.868/0001-57) Vol.No.10, Pages:340-354January2024.
15. S. Kamara and T. Moataz, "Boolean searchable symmetric encryption with worst-case sub-linear complexity," EUROCRYPT 2017, 2017.
16. D. Cash, S. Jarecki, C. Jutla, H. Krawczyk, M.-C. Rosu, and M. Steiner, "Highly-scalable searchable symmetric encryption with support for boolean queries," in Advances in Cryptology, CRYPTO 2013, ser. Lecture Notes in Comput. Sci., vol. 8042, 2013, pp. 353–373.
17. Sathish Polu and Dr. V. Bapuji. "Analysis of DDOS Attack Detection in Cloud Computing Using Machine Learning Algorithm", Tuijin Jishu/Journal of Propulsion Technology, Vol. 44, No.5, Pages:2410-2418, ISSN:1001-4055, December2023. https://scholar.google.co.in/citations?view_op=view_citation&hl=en&user=6hPSwVgAAAAJ&citation_for_view=6hPSwVgAAAAJ:UebtZRa9Y70C
18. Z. Fu, X. Wu, C. Guan, X. Sun, and K. Ren, "Toward efficient multi-keyword fuzzy search over encrypted outsourced data with accuracy improvement," IEEE Trans. Inform. Forensics Security, vol. 11, no. 12, pp. 2706–2716, 2016.
19. Q. Wang, M. He, M. Du, S. S. Chow, R. W. Lai, and Q. Zou, "Searchable encryption over feature-rich data," IEEE Trans. Dependable Secure Computing, 2016.
20. Y. Zhang, J. Katz, and C. Papamanthou, "All your queries are belong to us: The

power of file-injection attacks on searchable encryption," in 25th USENIX Security '16, Austin, TX, 2016, pp. 707–720.

21. A. A. Yavuz and J. Guajardo, "Dynamic searchable symmetric encryption with minimal leakage and efficient updates on commodity hardware," in Int. Conf. Selected Areas in Cryptography. Springer, 2015, pp. 241–259.
22. P. Rizomiliotis and S. Gritzalis, "Oram based forward privacy preserving dynamic searchable symmetric encryption schemes," in Proc. 2015 ACM Workshop Cloud Computing Security Workshop. ACM, 2015, pp. 65–76.
23. E. Stefanov, M. Van Dijk, E. Shi, C. Fletcher, L. Ren, X. Yu, and S. Devadas, "Path oram: an extremely simple oblivious ram protocol," in Proc. 2013 ACM SIGSAC Conf. Comput. Commun. security. ACM, 2013, pp. 299–310.
24. R. W. Lai and S. S. Chow, "Forward-secure searchable encryption on labeled bipartite graphs," in Int. Conf. Appl. Cryptography Network Security. Springer, 2017, pp. 478–497.