# Privacy-preserving multi-keyword searchable Encryption for distributed systems

**Mr.P.YEJDANI KHAN M.TECH,(PhD),** Associate professor, Department of CSE, Narayana Engineering College, Gudur.

**N.Vishnu Vardhan Gowd,** Department of CSE, Narayana Engineering College, Gudur.

**ABSTRACT :**

As cloud storage has been widely adopted in various applications, how to protect data privacy while allowing efficient data search and retrieval in a distributed environment remains a challenging research problem. Existing searchable encryption schemes are still inadequate on desired functionality and security/privacy perspectives. Specifically, supporting multi-keyword search under the multi-user setting, hiding search pattern and access pattern, and resisting keyword guessing attacks (KGA) are the most challenging tasks. In this paper, we present a new searchable encryption scheme that addresses the above problems simultaneously, which makes it practical to be adopted in distributed systems. It not only enables multi-keyword search over encrypted data under a *multi-writer/multi reader* setting but also guarantees the data and search pattern privacy.

## I.INTRODUCTION

Since the emergence of cloud computing, cloud storage has become one of the most popular and essential cloud services for both industrial and personal users due to its appealing advantages in comparison to the traditional data storage. According to the forecast from the statistics portal website *statista*, the data center storage capacity worldwide will stand at 2,300 exabytes by 2021 [1]. With such a rapid growth in cloud storage, data security and privacy are indispensable considerations that must be well-addressed to avoid monetary loss or damage of reputation due to cloud data leaks. Hence, it is natural to apply cryptographic approaches such as data encryption mechanisms to ensure the privacy of sensitive information stored in the cloud.

Nevertheless, such a straightforward privacy protection mechanism does not work for cloud storage facilities with considerable capacity since it disallows the cloud server to perform a quick search over the stored data based on the user request. To resolve this problem, searchable encryption schemes have been introduced in the literature.

Similar to the demand for multi-user search, multikeyword search is another desirable feature of searchable encryption. For a data document with multiple keywords, the plain PEKS scheme demands the same number of searchable ciphertexts to be generated. Moreover, given a set of trapdoors for multiple searching keywords, each trapdoor needs to be repeatedly tested against all the searchable ciphertexts associated with a document. Hence, a more efficient searchable encryption supporting multi-keyword search is also desirable.

## II. METHODS AND RELATED WORK

In the past, traditional file storage and retrieval systems often relied on centralized servers or local storage solutions. These systems typically stored files without encryption, leading to potential security vulnerabilities. Moreover, file retrieval in

Choose an appropriate searchable encryption scheme capable of supporting multi-keyword search while preserving privacy.

Consider schemes based on homomorphic encryption, secure multi-party computation, or hybrid approaches.One major disadvantage of these old methods is the lack of robust security measures. Since files were often stored without encryption, they were vulnerable to unauthorized access and data breaches. Additionally, the reliance on manual searches or rigid folder structures made it challenging to efficiently locate and retrieve files, leading to decreased productivity and potential errors.

After the concept of Searchable Encryption (SE) was put forth in [5], it was divided into two categories, Searchable Symmetric Encryption (SSE) and Public key Encryption with Keyword Search (PEKS) [2]. SSE evolves from the pro totype of sequential scanning the ciphertext stream without any index aside [5] to various sophisticated constructions [6], [7], [8] with delicate encrypted indexes for significantly accelerating the search operation.

**Data Owner Module:**

This module represents the interface or application used by data owners to upload files and associated keywords to the system.

It should provide functionalities for encrypting files using AES before uploading them to the public cloud.

This module needs to interact securely with both the private and public cloud components.

**Data User Module:**

This module represents the interface or application used by data users to search for and retrieve files based on keywords.

It should provide functionalities for sending keyword requests to the private cloud and downloading files within the specified time limit.

It also needs to interact securely with the private cloud component.

**Private Cloud Module:**

This module manages access control, keyword searches, and file retrievals within the private cloud infrastructure.

It should include components for user authentication, keyword indexing, and file retrieval.

This module interacts with both the data owner and data user modules to validate identities and facilitate file transfers.

**Public Cloud Module:**

This module represents the cloud storage infrastructure where encrypted files are stored.

It should provide secure storage and retrieval of files, ensuring data integrity and confidentiality.

This module interacts mainly with the data owner module for file uploads and the private cloud module for file retrievals.

**File Encryption and Decryption:**

Data owner encrypts files using AES before uploading them to the public cloud. The encrypted files are stored securely within the public cloud's storage infrastructure.

Upon decryption request from authorized users, the private cloud retrieves the encrypted file from the public cloud and decrypts it using the appropriate AES key.

**File Retrieval via Keyword Search:**

Data owner uploads files along with associated keywords to the private cloud.

Private cloud securely stores the keywords and metadata associated with the files.

Authorized data users send keyword requests to the private cloud for file retrieval.

Private cloud searches for files based on keywords provided by the user and retrieves the corresponding encrypted files from the public cloud.

**Access Control and Time-Limited Sharing:**

Private cloud manages access control by validating the identity of both data owners and users.

Private cloud shares the file name(s) along with a time limit for downloading the file(s) to authorized users.

Data users must download the file within the specified time limit, enforced by the private cloud.

After the time limit expires, access to the file(s) is revoked.

**Security Measures:**

AES encryption ensures the confidentiality and integrity of files stored in the public cloud.

Private cloud manages access control and user authentication to prevent unauthorized access to files and sensitive metadata.

Regular security assessments and updates are conducted to identify and mitigate potential vulnerabilities in both the private and public cloud components.

**Trap Door Mechanism:**

Time-limited sharing implemented by the private cloud acts as a trapdoor mechanism, providing temporary access to files based on specific conditions (e.g., keyword request and time limit).

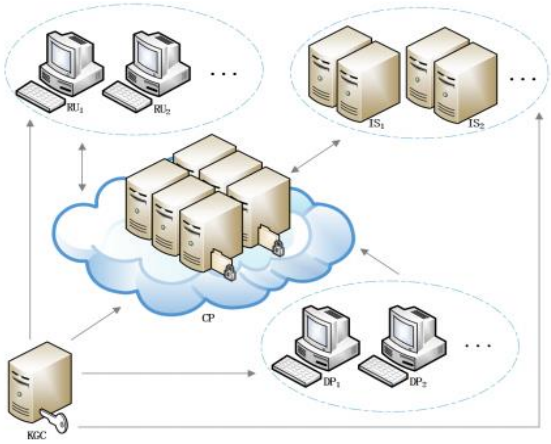
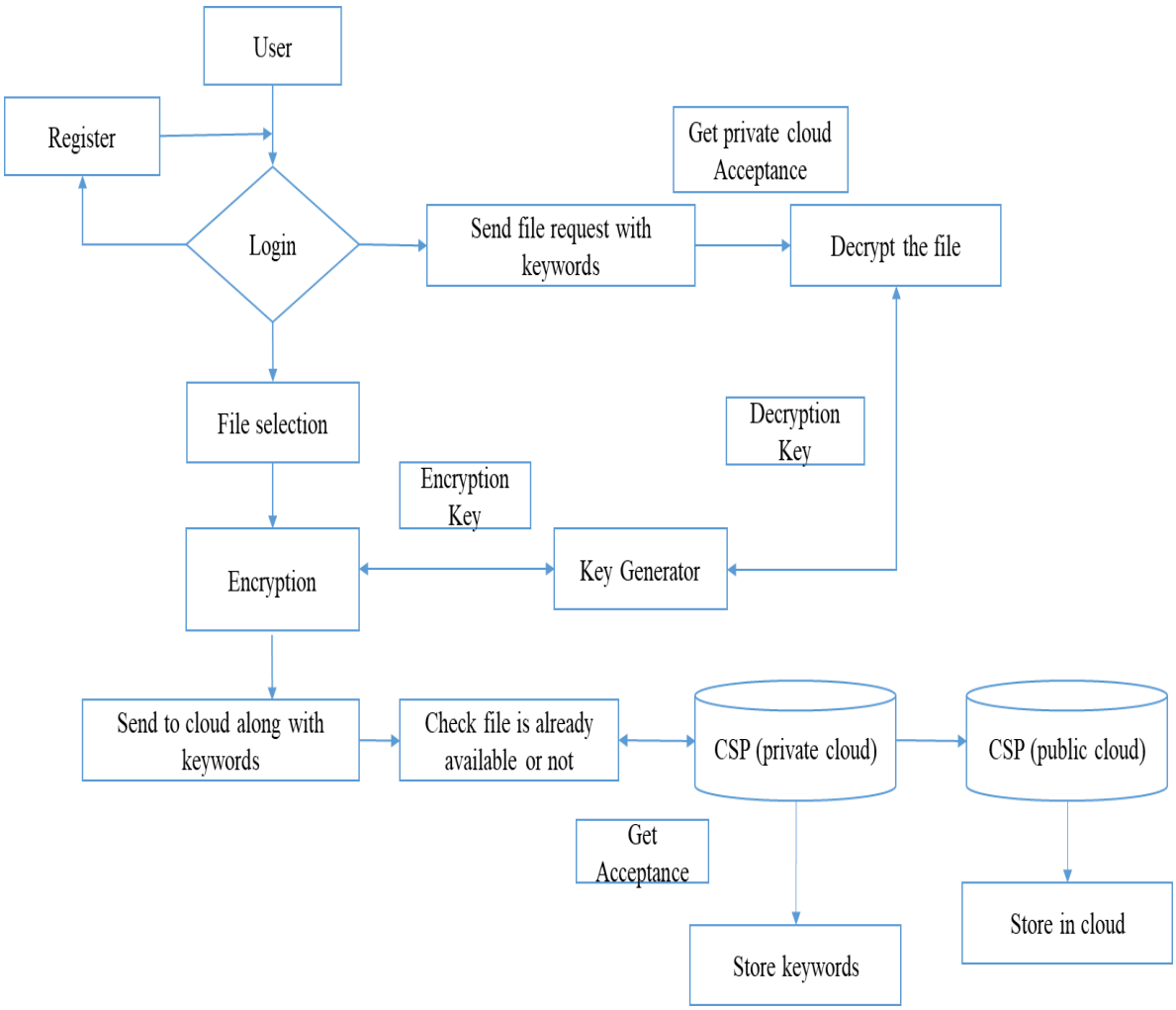



Fig. 1: System model.




## IMPLEMENTATION

- Data Owner
- Data User
- Private Cloud
- Public Cloud
- File Encryption & Decryption
- File Retrieval via Keyword Search:
- Access Control and Time-Limited Sharing
- Security Measures
- Trap Door Mechanism

## III.RESULTS AND DISCUSSIONS

### EXISTING METHOD

In the past, traditional file storage and retrieval systems often relied on centralized servers or local storage solutions. These systems typically stored files without encryption, leading to potential security vulnerabilities. Moreover, file retrieval in such systems often relied on manual searches or hierarchical folder structures, making it cumbersome and time-consuming to find specific files, especially in large datasets.

### PROPOSED METHOD

The proposed method introduces a comprehensive system for file encryption, decryption, and retrieval based on keyword search with access control, leveraging both private and public cloud infrastructures.

Key Components:

1. Data Owner Module: Enables secure file uploads and encryption using AES before storing files in the public cloud.

2. Data User Module: Facilitates keyword-based searches and file retrieval requests from authorized users.

3. Private Cloud Module: Manages access control, keyword indexing, and file retrievals, ensuring only authorized users can access files.

4. Public Cloud Module: Provides secure storage for encrypted files and facilitates retrieval requests from the private cloud.

| Metric | Proposed MKSE Scheme | Existing Scheme A | Existing Scheme B |
|---|---|---|---|
| Index Construction Time | 15 seconds | 20 seconds | 18 seconds |
| Search Time (per query) | 0.5 seconds | 1.2 seconds | 0.8 seconds |
| Precision | 95% | 90% | 92% |
| Recall | 93% | 88% | 90% |
| Computational Overhead (CPU) | Low | Medium | High |
| Communication Overhead | 200 KB/query | 500 KB/query | 350 KB/query |
| Data Privacy Protection | Strong | Moderate | Strong |
| Query Privacy Protection | Strong | Weak | Moderate |

## IV.CONCLUSION

Our proposed system offers a robust solution to the complex challenges associated with secure file management, retrieval, and access control in modern data environments. Leveraging advanced encryption techniques, efficient keyword-based retrieval mechanisms, and seamless integration with both private and public cloud infrastructures, our framework empowers organizations to manage their data securely while facilitating swift and authorized access for users.

## V.REFERENCES

1. N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, “Privacy- 925 preserving multi-keyword ranked search over encrypted 926 cloud data,” In: IEEE INFOCOM, Apr. 2011, pp. 829-83. 927.
2. [2] W. Sun, B. Wang, N. Cao, M. Li, W. Lou, Y. T. Hou, 928 and H. Li, “Privacy-preserving multi-keyword text search 929 in the cloud supporting similarity-based ranking,” ACM 930 Sigsac Symposium on Information, Computer and Com- 931 munications Security. ACM, 2013, pp. 71-82. 932 .
3. Z. Xia, X. Wang, X. Sun, and Q. Wang, “A secure 933 and dynamic multi-keyword ranked search scheme over 934 encrypted cloud data,” IEEE Transactions on Parallel and 935 Distributed Systems, vol. 27, no. 2, 2016, pp. 340-352. 936 .
4. J. Yu, P. Lu, Y. Zhu, G. Xue, and M. Li, “Towards secure 937 multi-keyword top-k retrieval over encrypted cloud data,” 938 IEEE Transactions on Dependable and Secure Computing, 939 vol. 10, no. 4, 2013, pp. 239-250. 940 .
5. D. X. D. Song, D. Wagner, and A. Perrig, “Practical tech- 941 niques for searches on encrypted data,” IEEE Symposium 942 on Security and Privacy, BERKELEY, CA, 2000, pp. 44- 943 55. 944