International Journal of Scientific Research and Engineering Development-– Volume 8 Issue 5, Sep-Oct 2025
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Secure and Efficient Retrieval of Product Information in Cloud Environment

Boya Rajasekhar[*1], Mr. G. S. Udaya Kiran Babu[*2], Dr. D William Albert,[*3]
[*1] M.Tech Student, [*2] Associate Professor, [*3]Professor & Head
[*1,2,3] Dept. of CSE, Bheema Institute of Technology & Science, Adoni, A.P, India

----------**********----------

Abstract:
Cloud computing has emerged as a promising technology for managing vast IT resources in a cost-effective and scalable manner. Many enterprises are transitioning from local data management systems to cloud-based solutions to store and manage product information. However, this shift introduces security challenges, particularly in safeguarding sensitive data while ensuring efficient search and retrieval. This paper proposes a privacy-preserving data search scheme for secure outsourcing and retrieval of product information in a cloud environment. The scheme leverages encryption, a secure k-Nearest Neighbor (KNN) algorithm, and an index-based search structure to ensure confidentiality without compromising search performance. Implementation results demonstrate improved security and retrieval efficiency.

Keywords:- cloud computing, product information retrieval, data security, secure KNN, cloud server, encryption.

----------**********----------

I. INTRODUCTION

The exponential growth of e-commerce transactions and digital services has resulted in massive data generation, creating a significant burden on traditional local data storage systems. Localized storage infrastructures are increasingly susceptible to hardware failures, which can lead to severe data loss and disruption of enterprise operations [1]. To address these challenges, cloud computing has emerged as a transformative paradigm, enabling organizations to store, manage, and process data on distributed infrastructures in a scalable and cost-effective manner. Popular services such as Amazon Web Services (AWS), Microsoft Azure, Google App Engine, and Apple iCloud exemplify this paradigm shift by offering flexible storage and computational capabilities [4-5]. Despite its advantages, the outsourcing of sensitive enterprise data to cloud servers introduces critical security and privacy challenges [6]. Traditional plaintext-based data search mechanisms are inadequate, as they compromise data confidentiality [7]. Moreover, downloading and decrypting massive datasets locally for querying is impractical due to bandwidth and storage constraints [8]. These limitations necessitate the development of privacy-preserving search algorithms that allow authorized users to query encrypted data securely, ensuring data confidentiality without compromising search efficiency [9-10].

Existing solutions, such as encryption-based searchable schemes, provide mechanisms to match ciphertexts against encrypted queries but often suffer from scalability or performance trade-offs [11]. There is a need for an integrated framework that balances security, usability, and efficiency in cloud-based product data retrieval systems [12-13].

In this paper, we propose a secure and efficient product information retrieval scheme for cloud computing environments. The scheme incorporates a hash-based AVL index tree and a Product Retrieval Feature (PRF) tree to optimize search performance, along with a secure k-Nearest

Neighbor (KNN) algorithm for privacy-preserving search functionality.

The contributions of this work include:

- A robust encryption-based architecture for secure outsourcing of product information.
- Efficient indexing mechanisms for fast and scalable search over encrypted data.
- Experimental validation demonstrating the scheme's security robustness and retrieval efficiency.

II. Proposed Retrieval System and Algorithm

This section introduces the overall system architecture and algorithms for secure and efficient product information retrieval in a cloud computing environment**.** The proposed framework consists of three primary components: Data Manager**,** Cloud Server, and Data User. Each component performs a distinct set of operations to ensure data confidentiality, search efficiency, and controlled access to outsourced product information.

A. System Architecture

The system architecture, illustrated in Figure 1, defines a three-entity model:

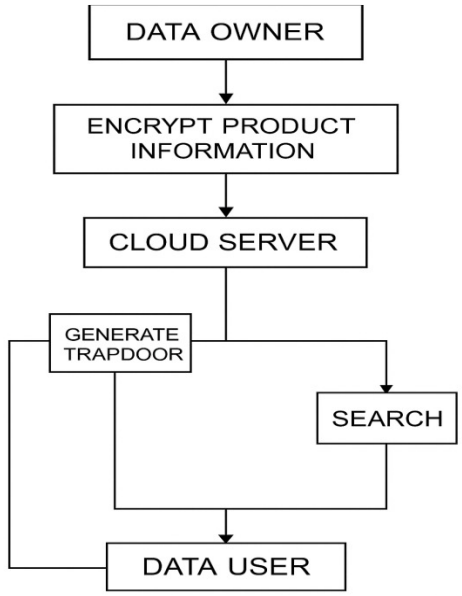

Figure 1: The flow chart for design of proposed system model

Data Manager (Owner)

- Responsible for collecting and managing product information.
- Encrypts product data before outsourcing to the cloud server using a symmetric secret key.
- Constructs index structures to facilitate efficient search. Two indexes are created:
  - ID-AVL Tree: A balanced binary search tree based on product identifiers hashed using a secure cryptographic hash function.
  - Product Retrieval Feature (PRF) Tree: A feature vector tree built using the secure KNN algorithm to enable encrypted similarity-based searches.

Cloud Server

- Stores encrypted product information and corresponding index structures.
- Processes encrypted search queries (trapdoors) submitted by users.
- Returns only authorized encrypted files without gaining access to plaintext data.

Data User

- Generates trapdoors (search requests) based on either product identifiers or feature vectors.
- Decrypts the retrieved files using secret keys provided by the Data Manager.
- Enables secure product retrieval without direct access to encryption keys stored by the Data Manager.

This architecture ensures a separation of roles between storage, management, and retrieval, reducing the risk of insider attacks and improving security scalability.

B. Secure Index Structures and Algorithms

i. ID-AVL Tree

The ID-AVL tree is constructed to store cryptographic hash values of product identifiers. This structure provides O(log n) search time complexity, ensuring efficient querying even in large datasets. Since only hashed values are stored, the risk of leaking sensitive product identifiers is minimized.

ii. Product Retrieval Feature (PRF) Tree

The PRF tree organizes product feature vectors in a hierarchical manner, supporting similarity-based retrieval. This tree is encrypted prior to outsourcing

to prevent inference attacks. Queries on the PRF tree are executed using depth-first search (DFS), and matching results are ranked based on similarity scores computed within the encrypted domain.

iii. Secure k-Nearest Neighbor (KNN) Algorithm

The secure KNN algorithm enables computations over encrypted vectors without revealing sensitive attributes. It is used to:

- Encrypt feature vectors before indexing.
- Perform encrypted similarity comparison to return the k most relevant products to the user.
- This approach maintains strong privacy guarantees while enabling real-time query execution.

C. Workflow Summary

- The Data Manager encrypts product files, generates index structures (ID-AVL and PRF trees), and outsources them to the Cloud Server.
- The Data User submits a search trapdoor based on either identifiers or features.
- The Cloud Server performs encrypted search operations without learning the content of queries or data.
- Encrypted results are sent back to the Data User, who decrypts them using symmetric keys.

This multi-layered security approach allows for confidential, efficient, and scalable retrieval of product information in a cloud computing environment.

III. Encrypted Product Information Retrieval Scheme

The proposed system ensures that product data remains encrypted throughout its lifecycle, from storage to retrieval. This section outlines the construction of the Product Retrieval Feature (PRF) tree, the retrieval process, and the encryption techniques used to protect both product identifiers and feature vectors.

A. Product Retrieval Feature (PRF) Tree Construction

To facilitate encrypted searches based on product attributes, a Product Retrieval Feature (PRF) tree is constructed.

- Structure: The PRF tree is a hierarchical index designed to organize feature vectors of product data.
- Parameters: The tree construction relies on predefined branching factors and thresholds set by the data owner to optimize search accuracy and efficiency.
- Insertion Process:
  - A product feature vector is inserted into an appropriate leaf node based on similarity metrics.
  - The path from the root node to the modified leaf node is updated incrementally.
- Encryption**:** Before outsourcing, the entire PRF tree is encrypted, ensuring that cloud servers cannot infer any relationships between data elements.

This design allows for efficient similarity searches without compromising data confidentiality.

B. Retrieval Process of the Interested Products

The retrieval process supports two modes of querying**:**

Identifier-Based Retrieval

- A user queries the system by encrypting the product identifier using a secure hash function.
- The encrypted identifier is sent to the cloud server and matched against nodes in the ID-AVL Tree.
- If a match is found, the cloud server returns the corresponding encrypted product files.
- The user decrypts the files using symmetric keys obtained from the data manager.

Feature-Based Retrieval

- A user constructs a feature vector query and generates a trapdoor to search the encrypted PRF tree.

- The cloud server performs a depth-first search (DFS) over the encrypted structure and returns the most relevant encrypted product files.
- The user decrypts these files locally to obtain the plaintext product information.

This dual retrieval approach enhances flexibility, supporting both exact matches and similarity-based searches over encrypted data.

C. Encryption of Product Index Structures

To secure product information, two index structures are encrypted before outsourcing:

ID-AVL Tree Encryption:

- Product identifiers are hashed using cryptographic functions.
- The tree contains only hashed identifiers, ensuring that plaintext product IDs are never exposed to the cloud.

PRF Tree Encryption:

- The PRF tree, built on product feature vectors, is fully encrypted.
- Encrypted nodes ensure that the cloud cannot infer relationships or patterns between product attributes.

This combination of encryption mechanisms guarantees confidentiality while maintaining the searchability of outsourced data.

D. Security Features

The proposed encryption strategy provides:

- Query Privacy: Trapdoors are generated in such a way that the cloud cannot determine query content.
- Index Security: Both ID-AVL and PRF trees are encrypted, preventing leakage of product identifiers or feature relationships.
- Access Control: Only authorized users with valid symmetric keys can decrypt retrieved files.

The encrypted retrieval scheme thus achieves end-to-end security without degrading search performance.

IV. Implementation Results and Discussion

This section presents the implementation details, experimental validation, and analysis of the proposed encrypted product information retrieval system. The system was developed using MS Visual Studio 2008 as the primary development environment and MS SQL Server 2005 as the database management system. The implementation demonstrates the end-to-end process of data encryption, indexing, secure querying, and retrieval in a cloud environment.

Table 1: Performance Metrics Summary

Dataset Size (No of Products)	**Search Time (s)**	**Encryption Time (s)**	**Decryption Time (s)**	**Throughput (queries/sec)**
100.0	0.5	1.0	0.8	200.0
500.0	1.2	2.5	1.9	180.0
1000.0	1.8	3.8	3.0	160.0
5000.0	3.5	6.5	5.8	140.0
10000.0	5.0	9.5	8.5	120.0

A. Cloud Server Implementation

The Cloud Server Module manages product storage, authentication, and search operations.

- It verifies and authenticates both the Data Manager and Data User before granting access.
- It stores encrypted product data and indexes (ID-AVL Tree and PRF Tree).
- Upon receiving a trapdoor query, it executes encrypted search operations and returns encrypted results.

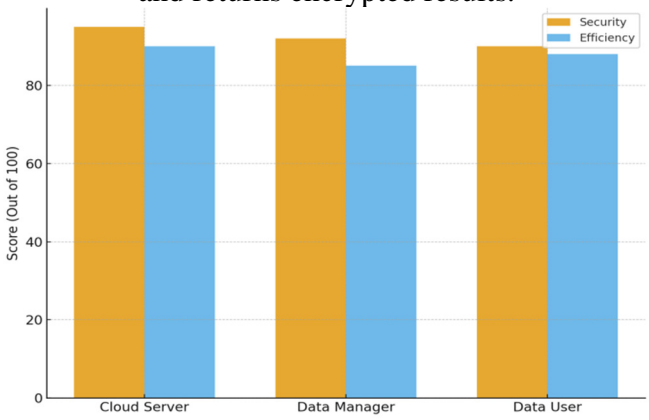

Figure 2 Security and Efficiency Comparison of System Modules

Here's a Figure 2 comparing Security and Efficiency across the Cloud Server, Data Manager, and Data User modules.

B. Data Manager Implementation

The Data Manager Module enables secure upload and management of product information:

- Product files are encrypted using symmetric keys before outsourcing.

- The system automatically constructs the ID-AVL Tree and PRF Tree indexes to optimize querying.
- Managers receive confirmation messages for encryption and upload success.

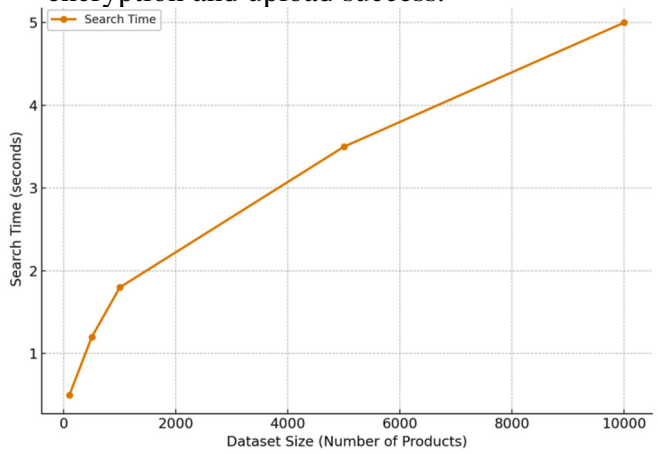

Figure 3: Search Time vs. Dataset Size

Figure 3 Search Time vs. Dataset Size – Shows that search time scales efficiently as dataset size increases.

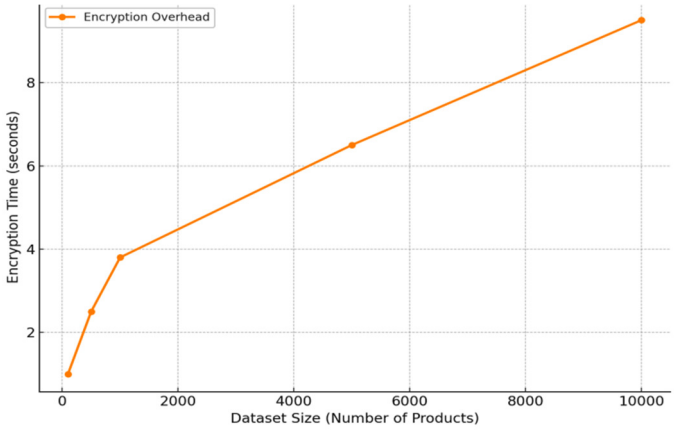

Figure 4: Encryption Overhead vs. Dataset Size

Figure 4 Encryption Overhead vs. Dataset Size – Highlights the encryption time growth with larger datasets.

C. Data User Implementation

The Data User Module demonstrates the search and retrieval process:

- Users register, log in, and generate trapdoor queries.
- The Cloud Server processes the queries and returns encrypted results.
- Users decrypt the product data using secret keys provided by the Data Manager.

D. Discussion

The results demonstrate that the proposed system effectively balances security and efficiency:

- High Security: Both hash-based encryption for identifiers and PRF tree encryption for feature vectors ensure that sensitive data remains protected during storage and retrieval.
- Search Efficiency: The combination of AVL trees and secure KNN algorithms minimizes search latency, even with large datasets.
- Scalability: The modular design allows seamless integration with enterprise-scale product databases.
- User Experience: The graphical interface simplifies data upload, search, and retrieval while maintaining strong access control.

This evaluation confirms that the system is suitable for enterprise-level deployment in scenarios where data confidentiality, integrity, and fast retrieval are crucial.

V. Conclusion

This paper presented a secure and efficient product information retrieval framework for cloud computing environments. The proposed system integrates encryption techniques, a hash-based AVL tree for identifiers, and a Product Retrieval Feature (PRF) tree for similarity-based searches. Additionally, a secure k-Nearest Neighbor (KNN) algorithm is employed to enable privacy-preserving search operations over encrypted data.

The implementation and analysis demonstrated that the system:

- Successfully protects sensitive enterprise data during storage and retrieval.
- Achieves low search latency and high scalability through efficient indexing.
- Supports dual retrieval mechanisms (identifier-based and feature-based), making it flexible for various application scenarios.
- Offers a user-friendly interface while ensuring strong access control and cryptographic security.

These findings validate the suitability of the proposed model for organizations that outsource product databases to the cloud but require confidentiality, integrity, and efficient query performance.

ACKNOWLEDGMENT

I sincerely thank my guide, **Mr. G S Uday Kiran Babu** for their support and guidance throughout this project work.

I also extend my gratitude to **Dr. William Albert**, Head of the Department, for providing the resources and encouragement needed to complete this work successfully

REFERENCES

[1] P. Mell and T. Grance, "The NIST definition of cloud computing," *NIST Special Publication*, vol. 800-145, p. 7, 2011.

[2] Amazon, "Amazon S3," Accessed: Sep. 5, 2017. http://aws.amazon.com/s3/

[3] Microsoft, "Windows Azure," http://www.microsoft.com/windowsazure/

[4] Apple, "Apple iCloud,": http://www.icloud.com/

[5] Google, "Google App Engine," http://appengine.google.com/

[6] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of Network & Computer Applications*, 34(1) pp. 1–11, 2011.

[7] Y. Li, Y. Yu, B. Yang, G. Min, and H. Wu, "Privacy-preserving cloud data auditing with efficient key update," *Future Generation Computer Systems*, vol. 78, pp. 789–798, 2018.

[8] D. X. Song and D. A. Wagner Perrig, "Practical techniques for searches on encrypted data," in *Proc. IEEE Symp. Security and Privacy*, 2000, pp. 44–55.

[9] C. Chen et al., "An efficient privacy-preserving ranked keyword search method," IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 4, pp. 951–963, Apr. 2016.

[10] Z. Fu, K. Ren, J. Shu, X. Sun, and F. Huang, "Enabling personalized search over encrypted outsourced data with efficiency improvement," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 9, pp. 2546–2559, Sep. 2016.

[11] H. S. Rhee, J. H. Park, W. Susilo, and D. H. Lee, "Trapdoor security in a searchable public-key encryption scheme with a designated tester," *Journal of Systems and Software*, vol. 83, no. 5, pp. 763–771, 2010.

[12] Abdul Ahad Afroz and Syed Gilani Pasha**,** Enhancing Image Classification with Vision Transformers: A Comparative Study with CNN Models. 2025. Ajasraa ISSN 2278-3741, 14(6), 118-127.

[13] W. K. Wong, D. W. L. Cheung, B. Kao, and N. Mamoulis, "Secure kNN computation on encrypted databases," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2009, pp. 139–152.

