OPEN ACCESS INTERNATIONAL JOURNAL OF SCIENCE & ENGINEERING

EFFICIENT MULTI KEYWORD SEARCH OVER ENCRYPTED DATA

Vijay B. Purohit[1], Mr. Md. Aijaz Ahmed [2]

ME (CNIS) MGM COE NANDED [1]

Asst. Professor, Department of Computer Science and Engineering MGM COE NANDED[2]

ABSTRACT: **Cloud computing are more popular for storing information by using storing statistics. By storing data into the cloud save the economical funds and gives the great flexibility. Data safety is the most challenging for the researchers. Confidential information must be encoded before outsourcing the data. Hierarchical clustering method presented in this paper used to the privacy preserving powerful search over encoded cloud data. This search over outsourced information, security is considered as a user revocation approach. Another more important component of this framework is the data duplication and it checking using SHA1 hashing strategy. Strategy will not allow saving the replica data at cloud server. EM clustering uses for clustering process and compare the EM clustering algorithm with K-means clustering algorithm. Experimental results state that the framework has various advantages like time utilization, efficient, easy and statistics duplication checking.**

Keywords- *Cloud computing, cipher text search, ranked search, multi-keyword search, hierarchical clustering, big data, and security.*

---∴∴∴---

I INTRODUCTION

Now a days cloud computing becomes more interesting methodology in different applications like academics and the industries. Cloud computing has a number of features, for example resource management, economical cost and simple and quick deployment [1]. Due to the tremendous economic advantages cloud computing, most of the organization deployed their cloud centers. Such cloud centers are Elastic Compute Cloud of Amazon, the App Engine of Google, the Azure of Microsoft, and Blue Cloud of IBM. Despite the fact that such lot of advantages, clients have a few stresses over outsourcing their information on cloud servers. Owners data is sometime sensitive information, for example, personal records, financial records pass record and so on., because once the data is outsourced, owner can not directly control the data. It is accessible online at some point. The Cloud Service provider (CSP) can keep up the security of such sensitive data by utilizing a few methods like fireware, virtualization, and Intrusion Detection System (IDS). For this situation, CSP increases full control over such information. But there is no guarantee or full trust on the representative of CSP [1] [2]. They can leak or alter the information. That is they can uncover the sensitive data of data owners. So to defeat the issue of security of outsource information, information encryption is one of the best solutions.

Cloud service provider can hold up the safety of important records by using the method like virtualization, and fire ware, Intrusion Detection System. Encryption system maintains the security of data. Download and decrypt the data from remote cloud server before a search happens, its illogical for the customer. Available encryption strategies made to provide the ability improving the encrypted reports via a key-word search [8][9].

Proposed system includes following points:

- Propose a clustering method to solve problem of maintaining the close relationship between different plain documents over an encrypted domain.
- Proposed the MRSE-HCI architecture to speed up server side searching phase.
- Design a search strategy to improve the rank privacy.
- Provide a verification mechanism to assure the correctness and completeness of search results.
- Generate ranked top-k documents.
- Provide user revocation method for security.

  To achieve above features and for better performance of system, we have used following algorithms and techniques in system.
- AES encryption algorithm and User revocation method for Security.

- EM clustering algorithm to enhance the clusters generation accuracy.
- Hierarchical algorithm to generate clusters hierarchically.
- SHA-1 algorithm for hashing
- Perform the data duplication operation for checking the duplicate data.

II REVIEW OF LITERATURE

Chen, Chi, et al [1], a hierarchical clustering strategy is proposed to support more search semantics furthermore to meet the demand for fast cipher text search within a big data environment. The proposed hierarchical approach clusters the documents based on the minimum relevance threshold and then partitions the resulting clusters into sub-clusters until the constraint on the maximum size of cluster is reached. In the search stage, this methodology can achieve a linear computational complexity against an exponential size increment of document collection. All together to confirm the authenticity of search results, a structure called minimum hash sub-tree is designed.

Cao et al. [2], interestingly, authors characterize and tackle the challenging issue of privacy-preserving multi-keyword ranked search over encrypted information in cloud computing (MRSE). They build up a set of strict privacy requirements for such a secure cloud information usage framework. Among different multi-keyword semantics, they choose the proficient similarity measure of "coordinate matching," i.e., whatever number matches as possible, to capture the relevance of data documents to the search query. They further utilize "inner product similarity" to quantitatively assess such similarity measure. They first propose a fundamental thought for the MRSE based on secure inner product computation, and after that give two essentially enhanced MRSE plans to accomplish different stringent privacy requirements in two distinctive threat models.

Wang et al. [3], characterize and solve the issue of secure ranked keyword search over encrypted cloud data. Ranked search enormously upgrades framework usability by enabling search result relevance ranking instead of sending undifferentiated results and further guarantees the file retrieval accuracy. In particular, they investigate the statistical measure approach, i.e., relevance score, from data recovery to manufacture a secure searchable index, and build up a one-to-many order-preserving mapping strategy to appropriately secure those sensitive score data. The resulting design is able to facilitate efficient server-side ranking without losing keyword privacy.

Chen et al. [4], a hierarchical clustering technique for cipher text search within big data environment is proposed. The proposed approach clusters the documents based on the minimum similarity threshold, and after that segments the resultant clusters into sub-clusters until the constraint on the maximum size of cluster is reached. In the search phase, this methodology can achieve a linear computational complexity against exponential size of document collection. In addition, retrieved documents have a better relationship with each other than traditional methods.

Various well-known [5] authentication protocols are considered with context of next generation mobile and CE network services. The potential weaknesses of current protocols can are overcome utilizing Zero Knowledge Proof (ZKP) strategies to ensure client passwords so an alternative ZKP protocol, SeDiCi 2.0, is depicted. This offers mutual and also two-factor authentication that is viewed as more secure against different phishing endeavors than existing trusted outsider protocols. The suitability of such a ZKP protocol for different CE-based cloud computing applications is illustrated.

Sun et al. [6], authors present a privacy-preserving multi-keyword text search (MTS) theme with similarity-based ranking to address this issue. To support multi-keyword search and search result ranking, they propose to create the search index based on term frequency and also the vector area model with cosine similarity live to attain higher search result accuracy. To enhance the search efficiency, they propose a tree-based index structure and numerous adaption ways for multi-dimensional (MD) algorithmic rules the sensible search efficiency is way higher than that of linear search. To further enhance the search privacy, they propose two secure index schemes to fulfill the stringent privacy needs under strong threat models, i.e., known cipher text model and known background model.

David Cash, Joseph Jaeger, Stanislaw Jarecki, Charanjit Jutla, Hugo Krawczyk, Marcel-C˘at˘alin Ro¸su, and Michael Steiner [7], they design and implement dynamic symmetric searchable encryption schemes that efficiently and privately search server-held encrypted databases with tens of billions of record-keyword pairs. The basic theoretical construction supports single-keyword searches and offers asymptotically optimal server index size, fully parallel searching, and minimal leakage. The implementation effort brought to the fore several factors ignored by earlier coarse-grained theoretical performance analyses, including low level space utilization, I/O parallelism and good put. They accordingly introduce several optimizations to our theoretically optimal construction that model the prototype's characteristics designed to overcome these factors. All of schemes and optimizations are proven secure and the information leaked to the untrusted server is precisely quantified. They evaluate the performance of our prototype using two very large datasets: a synthesized census database with 100 million records and hundreds of keywords per record and a multi-million webpage collection that includes Wikipedia as a subset.

Pang et al. [8], present a privacy-preserving, similarity-based text retrieval scheme that prevents the server from precisely reproducing the term composition of queries and documents, and anonymize the search results from unauthorized observers. In the meantime, their plan preserves the relevance-ranking of the search server, and empowers accounting of the number of documents that every client opens. The effectiveness of the scheme is verified empirically with two real text corpora.

S. C. Yu, C. Wang, K. Ren, and W. J. Lou [9], this paper addresses this challenging open issue by, on one hand, defining and enforcing access policies based on data attributes, and, on the other hand, allowing the data owner to delegate most of the computation tasks involved in fine-grained data access control to untrusted cloud servers without disclosing the underlying data contents. They achieve this goal by exploiting and uniquely combining techniques of attribute-based encryption (ABE), proxy re-encryption, and lazy re-encryption. The proposed scheme also has salient properties of user access privilege confidentiality and user secret key accountability.

III. SYSTEM OVERVIEW

A. Proposed System

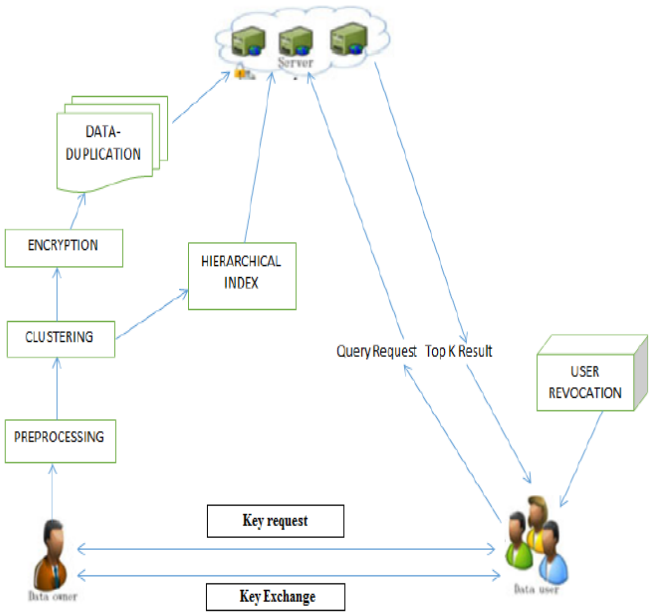

Figure 1: System Architecture

The propose system consist of three entities data owner, the data user and the cloud server. The data owner is responsible for collecting documents, building document index and outsourcing them in an encrypted format to the cloud server. Aside from that, the data user needs to get the authorization from the data owner before getting to the information. The cloud server gives a huge storage space and the computation resources required by cipher text search. Upon receiving a legal request from the data user, the cloud server searches the encrypted index, and sends back top-k documents that are most likely to match users query. The number k is appropriately chosen by the data user. This framework goes for protecting data from leaking information to the cloud server while improving the efficiency of cipher text search. In this model, both the data owner and the data user are trusted, while the cloud server is semi-trusted.

B. Algorithms

Algorithm 1: MRSE-HCI Architecture

Data owner creates encrypted index depending on the dictionary random numbers and secret key. User sends the query to cloud server for particular document. Cloud server returns documents to data user. Architecture contains the following algorithms.

- $\text{Keygen}\left(1^{l(n)}\right) \rightarrow (\text{sk}, \text{k})$: Used to generate the secret key to encrypt index and documents.
- $\text{Index}(\text{D}, \text{sk}) \rightarrow \text{I}$: Using secret key, encrypted index is generated in this phase. At the same time, clustering process is also included current phase.
- $\text{Enc}(\text{D}, \text{k}) \rightarrow \text{E}$: The document collection is encrypted by using symmetric encryption algorithm and which achieves semantic security.
- $\text{Trapdoor}(\text{w}, \text{sk}) \rightarrow \text{T}_\text{w}$: It generates encrypted query vector T_w with users input keywords and secret key.
- $\text{Search}\left(\text{T}_\text{w}, \text{I}, \text{k}_{\text{top}}\right) \rightarrow (\text{I}_\text{w}, \text{E}_\text{w})$: In this phase, cloud server compares trapdoor with index to get the top-k retrieval results.
- $\text{Dec}(\text{E}_\text{w}, \text{k}) \rightarrow \text{F}_\text{w}$ he returned encrypted documents are decrypted by the key generated in the first step.

Algorithm 2: K-means Clustering Algorithm

1. input the initial set of *k* cluster Centers *C*
2. set the threshold TH_{min}
3. **while** k is not stable
4. generate a new set of cluster centers C_Θ by k-means
5. **for** every cluster centers $C_{\Theta\lambda}$
6. get the minimum relevance score: *min* (S_i)
7. **if** the *min* $(S_i) < TH_{min}$
8. add a new cluster center: $k = k + 1$
9. **go** to while
10. **until** k is steady

Algorithm 3: Deduplication Checking

1. dbHash=getting files hash from the client side database which are successfully uploaded to the server
2. File Hash=users chunked file's hash
   H (New file) = h
   H (Old n chunks) = h_n
   Compare h and h_n
   If H (New chunk) == H (Old n files)

Chunk is duplicate and refuses it to store on public server

Else
Chunk is not duplicate and allowed to store on public server

Algorithm 4: SHA-1 Algorithm

Steps of SHA-1

Step 1: append padding Bits Message is "padded" with a 1 and as many 0's as necessary to bring the message length to 64 bitsfewer as an even multiple of 512.

Step 2: append length 64 bits are appended to the end of the padded message. These bits hold the binary format of 64 bits indicating the length of the original message.

Algorithm 5: User Revocation

Input: User Name Output: User terminated from system

Process:

Admin

1. Retrieve user List
2. Select user from the retrieve user list
3. Closed all operation from system (Selected User).
4. Then put that selected user in revoked list.

IV. RESULTS AND DISCUSSION

A. Experimental Result

In this section discussed the experimental result of the proposed system.

Time Comparison Graph for Clustering:

Following figure 2 shows the time comparison graph of the different clustering algorithm. Existing system used k-means algorithm for the clustering process, proposed system used the EM clustering algorithm for the clustering process. As from the following graph it is conclude that time required for completing the process of clustering by using the K-means algorithm is more than the EM clustering algorithm. Hence by using the EM clustering algorithm performance of the system is improved.

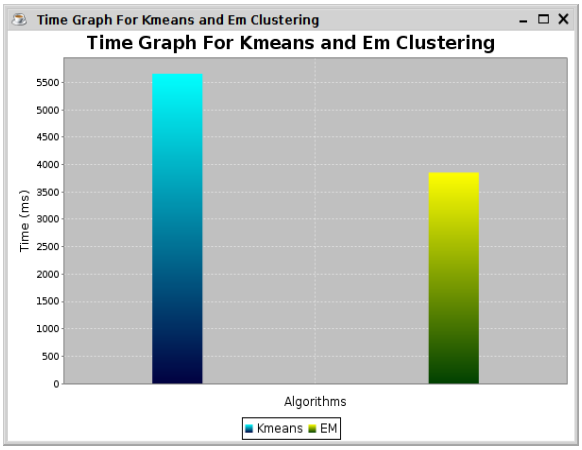

Figure 2. Time Comparison graph for clustering

Time Comparison Graph for Deduplication:

As if the duplicate file exists in the system, the performance of the system may be low, therefore in the proposed system we implement the concept of deduplication. In the following figure 3 shows the comparison of existing system without using the concept of deduplication and proposed system with using the concept of deduplication. From the figure it is conclude that time required for the system with deduplication is less than the time required for the system without deduplication.

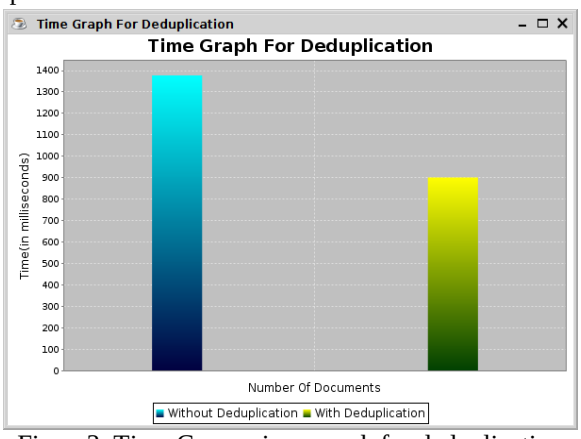

Figure3. Time Comparison graph for deduplication

Time Comparison Graph for Searching keyword:

The search accuracy can measure the client's fulfillment. The Retrieval accuracy is identified with two components: the relevance amongst reports and the queries and the relevance of documents between each other.

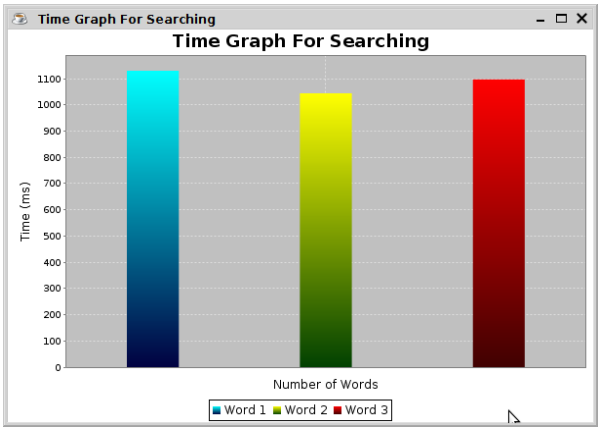

Figure 4. Time Comparison graph for searching keyword

Relevance of Document:

From the Fig.5, we can observe that the relevance of retrieved documents in the MRSE-with clustering is almost twice as many as that in the MRSE-HCI, which means retrieved documents generated by MRSE-with clustering are much closer to each other.

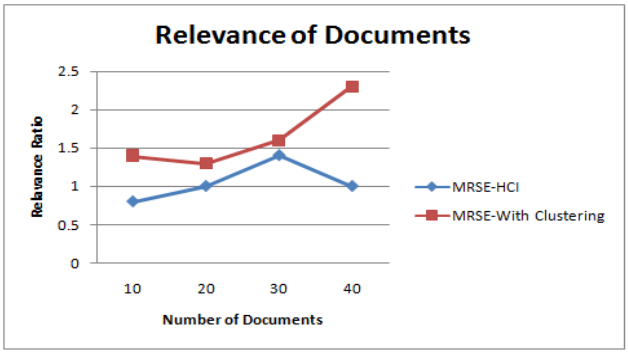

Figure 5: Relevance of Document

Search Time Graph

Fig 6 describes search efficiency using the different size of document set with unchanged dictionary size, number of retrieved documents and number of query keywords,

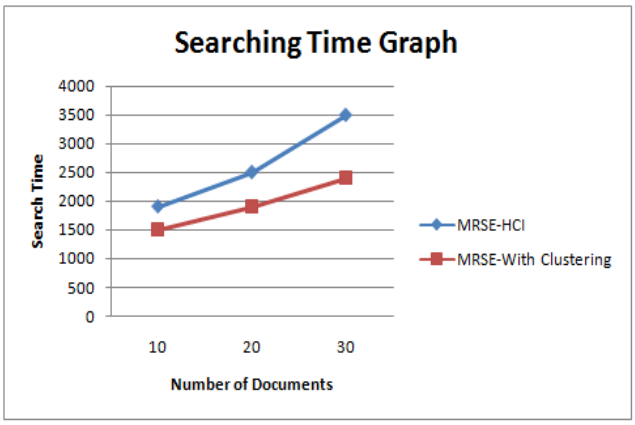

Figure 6: Search Time Graph

V. CONCLUSION

The systems taste cipher textual content search within the circumstances of cloud storage. Moreover speak the troubles of maintaining the semantic relationship between extra ordinary plain files over the associated encrypted files and provide the design technique to decorate the overall performance of the semantic search. The existing MRSE-HCI architecture adapts the requirements of data explosion, online information retrieval and semantic search. The experiment result proves that the proposed architecture not only properly solves the multi-keyword ranked search problem, but also brings and improvement in search efficiency, rank security, and the relevance between retrieved documents The proposed system enhance the system performance by implementing user revocation method where user group revocate, also system reduces the memory overhead and enhances searching speed by implementing data deduplication approach where duplicate data is removed.

REFERENCES

[1]. Chen, Chi, et al."An efficient privacy-preserving ranked keyword search method."IEEE Transactions on Parallel and Distributed Systems 27.4 (2016): 951-963.

[2] N. Cao, C.Wang, M. Li, K. Ren, andW. J. Lou, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," in Proc. IEEE INFOCOM, Shanghai, China, 2011, pp. 829-837.

[3] C. Wang, N. Cao, K. Ren, and W. J. Lou, ``Enabling secure and efficient ranked keyword search over outsourced cloud data," IEEE Trans. Parallel Distrib. Syst., vol. 23, no. 8, pp. 1467-1479, Aug. 2012.

[4] C. Chen, X. J. Zhu, P. S. Shen, and J. K. Hu, "A hierarchical clustering method For big data oriented cipher text search," in Proc. IEEE INFOCOM, Workshop on Security and Privacy in Big Data, Toronto, Canada, 2014, pp. 559, 564.

[5] S. Grzonkowski, P. M. Corcoran, and T. Coughlin, "Security analysis of authentication protocols for next-generation mobile and CE cloud services," in Proc. IEEEInt Conf. Consumer Electron, 2011, Berlin, Germany, 2011, pp. 83-87.

[6] W. Sun, B.Wang, N. Cao, M. Li, W. Lou, Y. T. Hou, and H. Li, "Privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking,"inProc. 8th ACM SIGSAC Symp. Inform. Comput. Commun. Security, Hangzhou, China, 2013, pp. 71-82.

[7] David Cash, Joseph Jaeger, Stanislaw Jarecki, Charanjit Jutla, Hugo Krawczyk, Marcel-Catalin Ro¸su , and Michael Steiner, "Dynamic Searchable Encryption in Very-Large Databases: Data Structures and Implementation." IACR Cryptology ePrint Archive 2014 (2014): 853.

[8] S. C. Yu, C. Wang, K. Ren, and W. J. Lou, "Achieving Secure, Scalable, and Finegrained Data Access Control in Cloud Computing," in Proc. IEEE INFOCOM, San Diego, CA, 2010, pp. 1-9.

[9] H. Pang, J. Shen, and R. Krishnan, "Privacy-preserving similarity based text retrieval,"ACM Trans. Internet Technol., vol. 10, no. 1, pp. 39, Feb. 2010.

[10] Y. H. Hwang, and P. J. Lee, "Public key encryption with conjunctive keyword search and its extension to a multi-user system," in Proc. Pairing, Tokyo, JAPAN, 2007, pp. 2-22.