A SECURE, DYNAMIC AND EFFICIENT MULTI-KEYWORD RANKED SCHEME OVER DATA ON ENCRYPTED CLOUD

Arati Deshmukh [1], Dr. S.T. Singh [2]

[1] Student, Department of Computer Engineering, PK Technical Campus, Chakan
[2] Principal, Department of Computer Engineering, PK Technical Campus, Chakan

ABSTRACT

New thing in cloud computing have modified the perspective of present day data innovation which is persuades the information proprietors to outsource the information to open cloud server for quick access to information its administration at least cost. Prior it was unrealistic to transfer the encoded information over the cloud. To give security the archive ought to be initially encrypted before outsourcing it and it can be retrieve successfully. Some more extra features dynamic update operations. For the index creation and query generation the vector space model and the TF IDF model are combined. A tree-based record structure is built and a GDFS algorithm is proposed to deliver multi-keyword ranked search. kNN calculation is utilized for encrypting the index and query vectors, to gain exactness of precision score between encrypted index and query vectors. To keep from statistical attacks, phantom terms are incorporated to index vector for building the search lists.

Keyword: - *cloud computing, multi-keyword ranked search, Searchable encryption*.

1. INTRODUCTION

Cloud computing is these days a broadly utilized innovation for giving ready services over the network. But as the use of cloud storage is expanding, the security chance, incorporation of information, and its secrecy are additionally verifiably expanding. In this way the cloud service providers (CSP) ought to oversee security and privacy, as they are assuming a fundamental part in data sharing functionality. The uncommon care ought to be taken look after the information security, as it is putting away at cloud storage which is overseen by outsider. As the use of web is expanding, the users likes to store or transfer information on cloud with the goal that they can get to the data from anyplace on the world. In any case, remembering the security traditional data storage techniques for validation are not that much reliable. For the insurance of the data over cloud, the data should be encrypted before transferring them to cloud to stay away from the acceleration which may open up with the classification of data.

Cloud storage is the critical and broadly utilized cloud computing model where information is stored on remote servers and managed and got to over internet. It is overseen and worked by the CSP on a server which promote data storage and is based on virtual machines. It works through the server virtualization which gives applications and data users a virtual design environment that is adaptable as per its necessities. . All the search schemes which support the multi - keyword functionality retrieve search output.

2. REVIEW OF LITERATURE

In [1] Wang et al used multi-keyword query. They choose the principle of “coordinate matching,” i.e., as many matches as possible, to capture the similarity between a multikeyword search query and data documents, and later quantitatively formalize the principle by a secure inner product computation mechanism. In [2], proposed search which solves processing overhead, data and keyword protection, least communication and computation overhead. The owner build index along with the keyword frequency-based relevance scores for files records. User asks for ’w’ to CS with optional ’k’ as Tw utilizing the private key. The CS seeks the index with scores and sends encrypted file in view of ranked sequence. The CS searches the index with scores and sends encrypted file based in light of ranked sequence. It does not play out multiple keyword searches and minimum overhead in index building.

Cao et al [3], proposed this search for known cipher content model and background demonstrate over encrypted data giving low calculation and communication overhead. An Efficient and privacy preserving in Multi-Keyword Ranked Search over encrypted cloud data the matching coordinate is decided for multi-keyword search. They utilized inner product likeness to quantitatively evaluate similitude for ranking files. The downside is that MRSE have small standard deviation _ which debilitates keyword security.

In [6], characterized and solved the issue of effective yet protected and sound rank keyword search over Encrypted cloud data. . Ranked search enormously upgrades system ease of use by giving back the matching files in a positioned arrange with respect to certain relevance criteria. In this way making one stage nearer towards sensible utilization of privacy preserving data facilitating services in Cloud Computing. These papers has characterized and solved the testing issue of privacy-preserving multi-keyword ranked search over encrypted cloud data (MRSE), and build up an arrangement of strict privacy necessities for such an ensured cloud data use system to wind up distinctly a reality. The proposed ranking method ends up being proficient to backtrack amazingly applicable documents relating to submitted search terms. Proposed ranking scheme is utilized as a part of our future system with a specific end goal to upgrade the security of data on Cloud Service Provider.

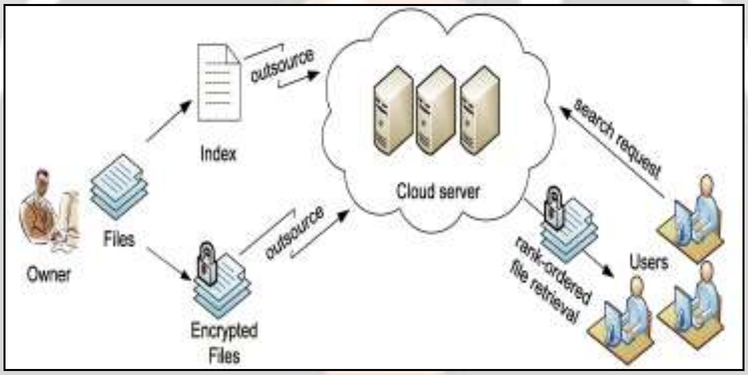


Fig 1: Architecture for search over encrypted cloud data

In [4], Obtainable searchable encryption scheme agree to a user to immovably search for over encrypted data through keywords without first decrypting it, these strategies support just conventional Boolean keyword search, without catching any relevance of the records in the output. At the point when straightforwardly connected in extensive joint data outsourcing cloud environment, they experience next shortcoming. Support just routine Boolean keyword search without decrypting it. Single-keyword search without ranking, Boolean - keyword search without ranking and Do not get applicable data.

The current strategies based on keyword-based information retrieval, which are broadly utilized on the plaintext data, can't be straightforwardly connected on the encrypted data. Downloading every one of the data from the cloud and decrypt locally is clearly illogical way. Keeping in mind the end goal to address the above issue, analysts have composed some universally useful arrangements with completely homomorphic encryption [4] or neglectful RAMs [5].

In [5], Boneh et al characterized the idea of a public key encryption with keyword search (PEKS) and giving two developments. Initial one is Constructing a PEKS is identified with Identity Based Encryption (IBE), however PEKS is seems to be harder to develop. They demonstrated that PEKS suggests Identity Based Encryption, yet the opposite is presently an open issue. Author’s developments for PEKS depend on late IBE developments. They could demonstrate security by exploiting additional properties of these plans.

In [7], Dan Boneh et al demonstrate to create a public-key encryption scheme for Alice that permits PIR looking over encrypted documents. In this they first to uncover no halfway data in regards to the user's search in the general public key setting and with non-insignificantly little correspondence small communication complexity. They gave a hypothetical answer for an issue postured by Boneh et al, "Open key Encryption with Keyword Search." The primary method of the arrangement likewise takes into account Single-Database PIR composing with sub-linear communication complexity, which consider of independent interest.

W Sun et al [8] described new techniques because remote searching over encrypted statistics the use of an untrusted server and gave confirmations to the security for the subsequent crypto frameworks. The systems which utilized was have various critical favourable circumstances: they are provably secure, they managed control then hidden search yet query isolation; those are simple and it proclaim almost no space and communication is overhead. Proposed scheme was permanency flexible, and then toughness without difficulty stay extended in imitation of support extra advanced search queries. Here conclusion was that they gives a capable new building obstruct for the development of secure administrations in the untrusted infrastructure.

Goh et al [9] developed a productive indcka secure list development called z-idx utilizing pseudo-random works What's more blossom filters, also utilize z-idx to actualize all the searches ahead encrypted data. This search plan might have been the majority effective encrypted data search scheme right now known; it furnished O(1) search time per document, What's more handles compacted data, variable length words, Furthermore Boolean and regular expression queries too .The strategies formed in this paper likewise utilized to build encrypted searchable review logs, private database query schemes, gathered hashing schemes, Furthermore secure set membership tests

M. Kuzu et al [10] proposed an efficient similarity searchable symmetric encryption scheme. They used area sensitive hashing which is utilized for quick similarity search as a part of high dimensional spaces for plain data. They proposed LSH based secure index and a search scheme to enables quick similarity search with regards to encrypted data. It's exceptionally basic not to give up the confidentiality of the sensitive information while giving functionality. They gave a rigorous security definition and demonstrated the security of the proposed scheme under the provided definition to guarantee the privacy. To clear up the properties of the proposed scheme, they displayed a real world use of it, to be specific error-aware keyword search, application enables keyword which tolerant to the typographical errors both in the queries and the data sources.

3. SYSTEM ARCHITECTURE

To provide effective multi-keyword ranked search on encrypted data over cloud supporting dynamic update operations and to minimize the search time by building the special tree-based index structure using "Greedy Depth-first Search" algorithm.

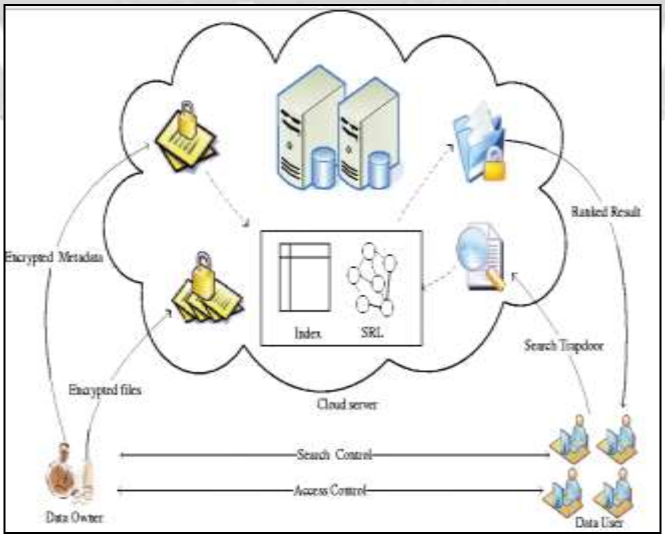


Fig 2: System Architecture

3.1 Defined Notations:

W - The dictionary, the set of keywords, denoted as W = w1; w2… wm
n - The total number in W
Wk - The subset of W, representing the keywords in the query
DC - Collection of n plaintext document, denoted as F = f1; f2… fn.
f- Document in the collection as a sequence of keywords.
d - Total number of documents in F.
C - Encrypted document collection stored in the cloud, denoted as C = c1; c2… cn.
T - Unencrypted form of index tree for the whole document collection F.
I - Searchable encrypted tree index generated from T.
Q - The query vector for keyword set Wq.
GT - The encrypted form of Q, which is named as trapdoor for the search request.
Iv - The index vector stored in tree node u whose dimension equals to the cardinality of the dictionary W.
u- A leaf node or an internal node of the tree.
Iu - The encrypted form of Iv.
Set Theory:-

Let M be the system to perform Multi Keyword Ranked Search over the Encrypted Cloud Data.
M= {I, O, F, Fail, Success}
Where, Inputs I = {U, K, E, C1, S, D}
U= User Login,
K= Extract Keywords,
E= Encryption of data using public key,
Cl= File and Keywords stored on cloud,
S= Search over the cloud data using the keywords,
D= Decryption of data

Where, O is an Output O = {A1, K2, E3, I4}
A1= Account Created,
K2= Keyword is Extracted in Encrypted form,
E3= Keywords are used for Encryption,
I4= Updating in the index,
D5= Encrypted data,

O6= Extraction of the original data.
Where, Functions set F

F1 {Setup},
F2 {Key Gen},
F3 {Encrypt},
F4 {Upload},
F5 {Keyword Search},
F6 {Decrypt}

Failure Conditions: For this situation, the search is not effective i.e., the user is not ready to locate the required document because there exists no matching keyword to the required query as required by the user.
Success Conditions: For this situation, the search is effective i.e., the user can locate the required document as the keyword exists in the file; the document is recovered by the user.

Graph Representation:-
Let G be a closed graph that represents our system; Such that G = {V, E }
Where,
E represents the set of edges; E = {e1, e2, e3 ....., e10} and V is a set of vertices; V = {v1, v2... v5}.
In graphical representation of system, vertices set, V represents the modules which are connected through the directed edges in set E which represents the input/output of modules.

Let r be a rule of E into V such that for a given edge;
r will return vertices. r (E) V.
Thus, for this system,
r(e1) = v2.......v2 is called using e1 for keyword generation and file encryption.
r (e2) = v3....... data is passed to v3 using e2 to upload a file.
r(e3) = v4...... e3 is used to pass data to v4 for encryption of the keywords.
r (e4) = v5...... e4 is used to upload the encrypted keywords and generating the index at v5.
r(e5) = v6......the edge e5 is passed to v6 to provide encrypted index for searching.
r(e6) = v7......the edge e6 is passed to v6 providing the encrypted keywords for searching.
r (e7) = v8....... the edge e7 is passed to v7 providing the unencrypted keywords provided by the user.

Algorithm:

Build tree-Index
Input: Document collection DC = {d1; d2... dn} with the identifiers ID = {ID|ID = 1; 2... n}.
Output: Index tree T

```
for each document dID in DC do
Construct leaf node v for dID, with v.ID = GenID (), v.Pl = v.Pr = null, v.ID = ID, and D[i] = TDCdID; wi for i = 1... m;
Add v to CurrentNodeSet;
end for
while number of nodes in CurrentNodeSet is greater than 1 do
if number of nodes in CurrentNodeSet is even, i.e. 2u then
for each pair of nodes v' and v" in CurrentNodeSet do
Generate parent node v for v' and v", with v.GID = user GenID(), v.Pl = v', v.Pr = v", v.ID = 0 and D[i] = maxfv'.D[i]; v".D[i]g for each i = 1;.....; m;
Add v to TempNodeSet;
end for
else
for each pair of nodes v' and v" of former (2u - 2) nodes in CurrentNodeSet
do
Create parent node v for v' and v";
Add v to TempNodeSet;
end for
Generate parent node v1 for the (2u - 1)-th and 2u-tu node, and then generate parent node v for v1 and the (2u + 1)-tu node;
Add v to TempNodeSet;
end if
Replace CurrentNodeSet with TempNode-Set and then clear TempNodeSet;
end while
return only node left in CurrentNodeSet, namely, root of index tree RT ;
```

Here, the data owner will encrypt the document utilizing the outside application. Alongside the encrypted document the Data Owner will get the keywords which he/she can use for the search reason. After that, the encrypted document gets transferred to the server. Encrypted keywords get transferred to the server moreover. Index will get created utilizing the build tree Index algorithm. End user can search the data. They can't see the encrypted data without the authorization of data owner. Demand will be sent to the owner that the accompanying user needs to get to the contents of the document. Data owner will then send the way to the user secretly through the mail. End user can change the substance of the file and save with different version.

The system consists of 4 main implementation steps:-
SK Setup (): Here system set the secret vector S as an n-bit vector, and set M1 and M2 are (n + n') (n+ n') invertible matrices, where m is the number of phantom terms.
I GenIndex (F; SK): Before encrypting index vector Iv, extend the vector Iv to be (n+n') - dimensional vector. Each extended element Iv [n+ i], i = 1... n', is set as a random number E (i).

GT GenTrapdoor (Wk; SK): Query vector Q is extended to be (n + n') - dimensional vector. Among the extended elements, a number of m elements are randomly chosen to set as 1, and the rest are set as 0.
RelevanceScore SRScore(Iu; GT): After the execution of relevance evaluation by cloud server, the final relevance score for index vector Iu equals to Iv.Q + ΣE(v), where v ε {j— Q[n + i]} = 1.

4. SYSTEM ANALYSIS

The proposed scheme is implemented using the private cloud setup with open stack. The Asp.Net and C# language is used for building the system pages that are used in demonstration of the proposed work. The tests include Search precision on various privacy levels; in this execution we have 3 principle modules, Cloud Server Module, Data Owner Module and Data User Module.

.

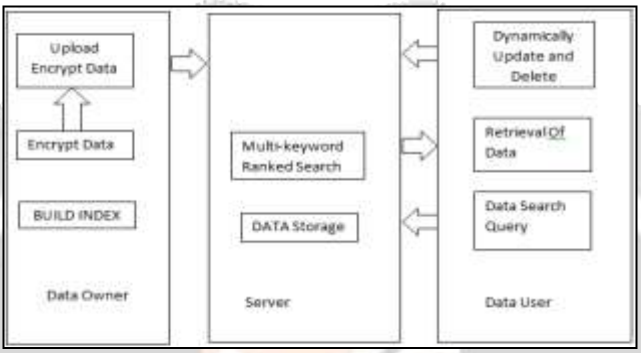
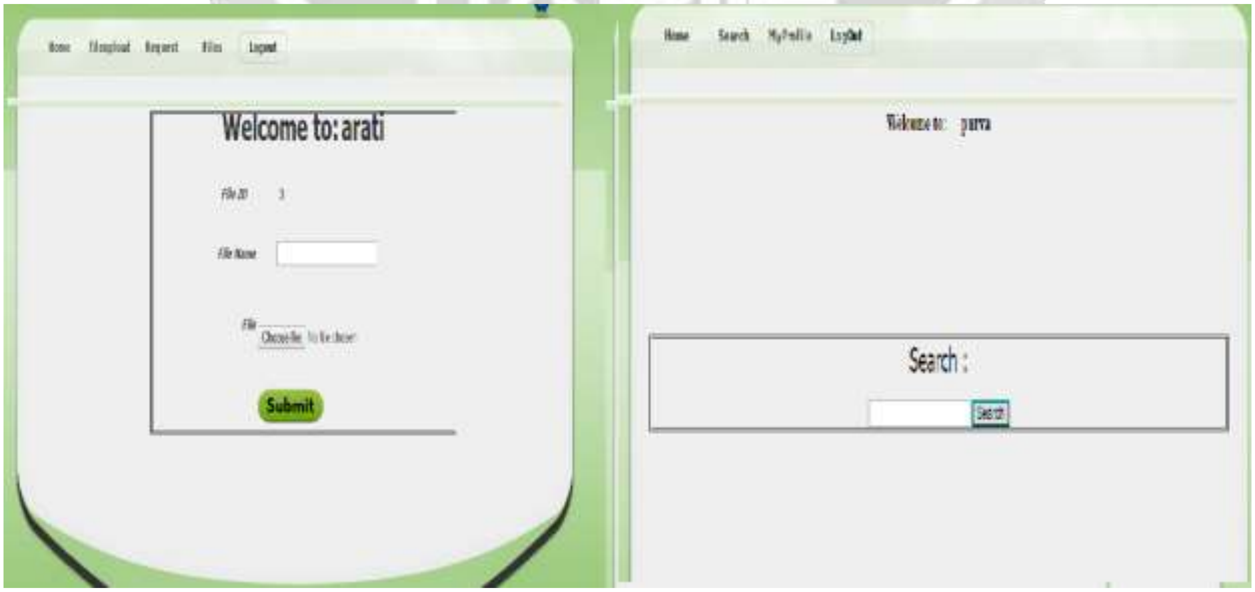


Fig 3: System Modules

In cloud, kept up by the cloud service providers, gives storage room to facilitating data files in a compensation as-you-go way. In any case, the cloud is untrusted since the cloud service providers are effectively to end up untrusted. Accordingly, the cloud will attempt to take in the content of the store data. In Data Owner assumes responsibility of system parameters era, user enlistment, and user repudiation. In the real applications, the group manager more often than not is the leader of the group. Along these lines, we expect that the group manager is completely trusted by alternate parties. In Group Member, Users are an arrangement of enlisted users that will store their own data into the cloud and impart them to others. In the plan, the group membership is powerfully changed, because of the new user enlistment and user renouncement.



Welcome to: arati

File ID 3

File Name

File Choose file No file chosen

Welcome to: purva

Search :

Fig 4: File upload and search in Cloud

The search precision of this system is influenced by the phantom keywords in proposed technique. Here, the precision is characterized as that in [5]: $Pk = k'/k$, where k' is the quantity of real k files as top ranked in the retrieved k files. On the off chance that a minor standard deviation is set for the random variable _v, this procedure should get larger precision, and vice-versa. The outcomes are appeared in Fig.2 (a). As said before here phantom terms are added to the index vector to change the relevance score calculation, so that the cloud server can't identify keywords by checking the TF circulations of special keywords. Here, we measure the obscures of the importance score by "rank privacy", which is characterized as:

$$P'k = \Sigma|ri - r'I \,| \,/\, k2;$$

5. CONCLUSION

The proposed technique for the multi keyword ranked search on encrypted data permits search as well as the dynamic updating data of those documents. Here, the data owner is additionally in charge of the updating data of the data required for the index generation. So the additional steps are required by the data owner each time he transfers the document to the cloud. Yet, there are many difficulties in a secure multi user system i.e. the data owner consider that every one of the users are trustworthy not not dishonest. Its impractical, a dishonest user may prompt too many secure issues like imparting the secure key to the unauthorized person or offering the decrypted information document to another association. In future works, we will attempt to finish the difficulties for the secure system to enhance its security also will try to improve the SE scheme to handle these challenges. There are still many challenges in symmetric SE schemes

6. REFERENCES

[1]. C. Wang, N. Cao, J. Li, K. Ren, and W. Lou, "Secure Ranked Keyword Search over Encrypted Cloud Data," Proc. IEEE 30th Int'l Conf. Distributed Computing Systems (ICDCS '10), 2010.
[2]. Cong Wang et al.,"Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data", IEEE Transactions on parallel and distributed systems, vol. 23, no. 8, August 2012.
[3]. Ning Cao et al.,"Privacy-Preserving MultiKeyword Ranked Search over Encrypted Cloud Data", IEEE Transactions on parallel and distributed systems, vol. 25, no. 1, Jan.
[4]. Madane S.A, "Comparison of Privacy Preserving Single- Keyword Search and Multi-Keyword Ranked Search Techniques over Encrypted Cloud Data", 2014 International Journal of Computer Applications (0975 - 8887) Volume 126 - No.14, September 2015.