MULTI KEYWORD RANKED SEARCH OVER ENCRYPTED CLOUD DATA

[1]T. SaiPrasad Reddy, [2]Ch. Bhaskar Rao, [3]A. Vamsi Krishna, [4]T. Vamsi Vardhan Reddy
[1]Associate Professor, [2] Assistant Professor, [3] Assistant Professor, [4] Assistant Professor
Dept. of CSE,
Narayana Engineering College, Nellore, A.P, India

Abstract- The project solves and defines the difficulty of multi-keyword ranked search over encrypted cloud data (MRSE) while preserving firm system wise privacy in the cloud computing hypothesis. Hence to protect privacy of the data, before privacy data also outsourced to the cloud data that has delicate to be encrypted, which make the valuable data utilization service not a easy task. Even though searchable encryption method allows users to firmly search over encrypted data all the way through the keywords, they carry only search i,e Boolean. They are not yet enough to meet the utilization of the data successfully because there is instinctively demanded by large number of data files and users located in cloud. Hence it is required to allow multiple keywords in the search request and return documents in the order of their significance to the keywords. The keyword i,e Boolean of the search technique only produce the unsorted result. An valuable method proposed for this difficult problem is privacy conserving search over encrypted cloud data. After the data has been encrypted and outsourced by the data owner this method establishes a set of privacy desires for secure cloud data utilization system during splitting the cloud data and storing the chunk data in different servers. Among different multi-keyword aetiology, this method chooses the well-organized similarity measure of "coordinate matching" for searching technique.

Keywords- *Cloud, Hypothesis, Boolean, Chunk data.*

I. INTRODUCTION

Cloud computing is an casually using the real-time communication network and connect large number of computers that depict different types of computing concepts. Non-ambiguous technical or scientific description in cloud computing has not been accepted. In science, cloud computing is a one kind of the distributed computing network and capability to run a program on many related computers at the similar time. Cloud computing is called as a utility of the computing since it uses pay per use paradigm. In cloud computing, users can also right to use a variety of resources like storage, programs, and application development platforms. cloud computing is an emerging technology and it is also called as utility because client are used to store their data in the cloud server. In cloud server data can also be leaked to hackers therefore encrypted the data before sent to the cloud for data privacy [1].

cloud computing is transform how businesses uses the information technology . Several trend are opening up the period of cloud computing, which is an development of Internet-based and use of technology of the computer. More controlling processors and increasingly cheaper , collectively with the Software as a Service computing architecture, are transforming data centres into pool of computing service on a huge scale. The escalating network bandwidth and reliable yet stretchy network connections make it even possible that users can now give to high quality services from data and software that dwell exclusively on remote data centres .To protect privacy of the data and struggle unwanted accesses in the cloud, Cloud service providers (CSP) frequently put in force user„s data security all the way through mechanisms like virtualization and firewalls. However, these techniques do not protect user„s privacy from the CSP itself since the CSP possesses full organize of the system hardware and minor levels of software stack. Therefore encryption before outsourcing the data of the cloud; this, however, Difficulty in the traditional data utilization service based on plaintext keyword investigate. The small Solution of downloading all the data and decrypting close by is clearly unreasonable, due to the vast amount of bandwidth cost in cloud extent systems. Thus, exploring privacy preserving and effective search service over encrypted cloud data is of dominant importance [2]. Considering the potentially large number of on-demand data users and enormous quantity of outsourced data documents in the cloud, this crisis is particularly challenging as it is enormously difficult in system usability, performance and scalability. On the one hand, to meet the effectual data retrieval, the huge amount of documents insists the cloud server to perform result significance ranking, instead of returning results which are not similar. Such ranked of the search system enables the data users to discover quickly the most appropriate information, rather than sorting all the way through each and every match in the content set. On the other hand, to progress the search result exactness as well as to increase the experience of user searching. To provide more exactness to the end Privacy Conserving In Cloud Documents in excess of users result is done by searching, the unlabelled data keywords are incorporated in the index of the server and then searching is done this search results is then characterized and then they are sorted in their splitting up using Top k query algorithm. TOP-k selection queries will assist to sort the related data and provide the accurate data to the end user [3]

II RELATED WORK

Organizations, companies store more and more valuable information is on cloud to protect their data from virus, hacking. The benefits of the new computing model include but are not limited to: relief of the trouble for storage administration, data access, and avoidance of high expenditure on hardware mechanism, software, etc. Ranked search improves system usability by normal matching files in a ranked order regarding to certain relevance criteria (e.g., keyword frequency),As directly outsourcing relevance scores will drips a lot of sensitive information against the keyword privacy, We proposed asymmetric encryption with ranking result of queried

III. EXISTING SYSTEM

The cloud server hosts intermediary information storage space and services can also be regain. Since information may hold susceptible information, defending information cannot be fully entrusted by cloud servers. For this reason, files that can be should be encrypted. Any information type that is leaking that would have an effect on data solitude is regard as in̆sufferable. To get mutually the effectual data recovery need, the huge quantity of documents stress the cloud server instead of recurring not similar results and to carry out result impact ranking . Such data users approaches ranked search system to discover the most appropriate information rapidly, rather than onerously sorting all the way through every match in the content set. Ranked of search remove the needless network traffic by distribution reverse only the most of the valid data, which is also very attractive in the "pay- as-you-use" cloud perception. For segregation support, such grade of the process, however, cannot be seep out any information related to the keyword. To get good search consequence correctness as well as to improve the user sharp skill, such ranking system essentially hold up multiple keywords search, too rude results was produced by single keyword of search [4].

Draw Backs of Existing System

Accurate data we are not getting.

Users will not get adequate required ranking functionality.

Data that can be sharing will not be safe.

IV PROPOSED SYSTEM

In the future work, we will discover checking the integrity of the rank order in the search consequence assume the cloud server is untrusted.

To suggest OTP (one Time Password) as our upcoming work. This OTP (One Time Password) used to observe data in cloud and it can be used only one occasion, when you looking for a file and be inclined to view the file the OTP will transmit to electronic message and you obtain the OTP and be relevant to see the file.

A. Data Flow Diagram

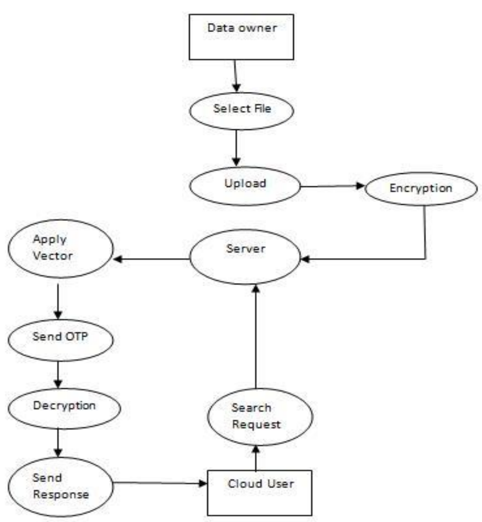

Fig.1 DataFlow Diagram

The cloud server hosts third-party data storage and retrieve services. Since data may contain sensitive information, the cloud servers cannot be fully entrusted in protecting data. For this reason, outsourced files must be encrypted. Any kind of information leakage that would affect data privacy may regard as unacceptable. The data owner has a collection of n files to outsource onto the cloud server in encrypted form and expects the cloud server to provide keyword retrieval service to data owner himself or other authorized users. To achieve this, the data owner needs to build a searchable index from a collection of keywords and then outsources both the encrypted index and encrypted files onto the cloud server as shown in the fig 1. The data user is authorized to process multi-keyword retrieval over the outsourced data. The computing power on user side is limited, which means that operations on user side should be simplified. The authorized data user at first generates a query. For privacy consideration, which keywords the data user has searched must be concealed. Thus the data user encrypts the query and sends it to the cloud server that returns the relevant files to the data user (e.g. Fig. 1). Afterwards, the data user can decrypt and make use of the files [5].

Advantages

Ranking based search for users are more convenient

Proposed cloud storage systems that provide confidentiality, integrity and verifiability of client data against un- trusted cloud provider.

B. System Architecture

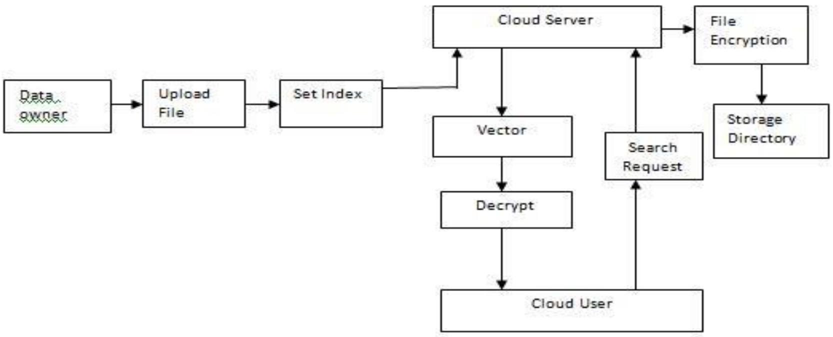

Fig.2 System Architecture

V. RANKED SEARCH

The multi-keyword search method checks whether queried keywords exist in a document or not. If the user searches for a single or more keywords, there will possibly be many correct matches where some of them may not be useful for the user at all. Therefore, it is difficult to decide as to which documents are the most relevant. I add ranking capability to the system by adding extra index information for frequently occurring keywords in a file. With ranking, the user can retrieve only the top τ matches where τ is chosen by the user. In order to rank the documents, a ranking function is required, which assigns relevancy scores to each document matching to a given search query. One of the most widely used metrics in information retrieval is the term frequency. Term frequency is defined as the number of times a keyword appears in a document. Instead of using term frequency itself, we assign relevancy levels based on the term frequencies of keywords. I assume that there are η levels of ranking in our pro- posed method for some integer $\eta \geq 1$. For each document, each level stores an index for frequent keywords of that document in a cumulative way in descending order. This basically means that ith level index includes all keywords in the (i + 1)th level and the keywords that have term frequency for the ith level. The higher the level, the higher the term frequency of the keywords is. For instance, if $\eta = 3$, level 1 index includes keywords that occur at least once in the document while levels 2 and 3 include keywords that occur at least, say 5 times and 10 times4, respectively. There are several variations for relevancy score calculations and we use a very basic method. The relevancy score of a document is calculated as the number representing the highest level search index that the query index matches. All the keywords that exist in a document are included in the first level search index of that document .The other higher level indices include the frequent keywords that also occur in its previous level, but this time they have to occur the number of times, which are at least the term frequency of the corresponding level. The highest level includes only the keywords that have the highest term frequency. In the oblivious search phase, the server starts comparing the user query against the first level indices of each document.

The matching documents found as a result of the comparison in the first level are then compared with the search indices in the other levels according to the Algorithm 1. In this method, some information may be lost due to the ranking method employed here. Rank of two documents will be the same if one involves all the queried keywords in- frequently and the other involves all the queried keywords frequently except one infrequent one. The rank of the document is identified with the least frequent keyword of the query. We tested the correctness of our ranking method by comparing with a commonly used formula for relevance score calculation, which is given in the following:

The number of levels and the term frequency of each level can be chosen in any convenient way.

```
Algorithm Ranked Search
{
 for all documents Ri do
 {
  Compare (level1 index of Ri , query index) j = 1 while match do
  {
   increment j
   Compare (levelj indices of Ri, query index) end while
  }
  rank of Ri = highest level that match with query index end for
 }
}
```

$$Score(W,R) = \sum_{t \in W} \frac{1}{|R|} \cdot (1 + \ln f_{R,t}) \cdot \ln\left(1 + \frac{M}{f_t}\right).$$

Here W, fR,t, ft, and M denote the set of searched key- words, the term frequency of term t in file R, the number of files that contain term t, and the number of files in the database, respectively. |R| is the length of the file R. We use a synthetic database to compare the two rank- ing methods. We assume that there are 1000 files of equal lengths in the database and 3 keywords are searched for. We further assume that the number of files containing the queried keywords (ft) is equal to 200 and only 20 of those contain all 3 keywords. Term frequencies of the keywords in the 20 matching files are chosen uniformly randomly between 1 and 15 and η = 5 in our proposed ranking method. In 40% of the time, the top match for a given relevance score , is also the top match for our proposed ranking method, and 100% of the time in the top 3 matches of our ranking method. Additionally in 80% of the time, at least 4 of the top 5 matches for the given relevance score is also in the top 5 of our proposed ranking method. Note that since we assume ft is the same for all t ∈ W, changing ft has no effect on the performance of both methods. As can be observed from these experimental figures, while the performance of the proposed method is inacceptable levels, the choice of the method parameters (especially η and term frequency of each level) depends very much on the characteristics of the database and the documents. While this new method necessitates an additional r-bit storage per level for a document, it reduces the communication overhead of the user since matches with low rank documents will not be retrieved unless the user requests.

Considering η search indices are stored instead of a single search index per document, storage overhead for indexing mechanism increases η times due to ranking. This additional cost is not a burden for the server since the index sizes are usually negligibly small compared to actual document sizes.

Algorithm Used

A .RSA Algorithm

This algorithm is used to encrypt n decrypt file contents. It is an asymmetric algorithm. The RSA algorithm involves three steps: key generation, encryption and decryption. Key generation RSA involves a public key and a private key. The public key can be known to everyone and is used for encrypting messages. Messages encrypted with the public key can only be decrypted using the private key. The keys for the RSA algorithm are generated the following way

1. Choose two distinct prime numbers a and b.
2. Compute n = ab. n is used as the modulus for both the public and private keys
3. Compute φ(n) = (a – 1)(b – 1), where φ is Euler's totient function.

4. Choose an integer e such that 1 < e < φ(n) and greatest common divisor of (e, φ(n)) = 1; i.e., e and φ(n) are co prime. e is released as the public key exponent. having a short bit-length .

B.K-Nearest Neighbor

K-nearest neighbor search identifies the top k nearest neighbors to the query. This technique is commonly used in predictive analytics to estimate or classify a point based on the consensus of its neighbors. K-nearest neighbor graphs are graphs in which every point is connected to its k nearest neighbors. The basic idea of our new algorithm: The value of dmax is decreased keeping step with the ongoing exact evaluation of the object similarity distance for the candidates. At the end of the step by step refinement, dmax reaches the optimal query range Ed and prevents the method from producing more candidates than necessary thus fulfilling the r-optimality criterion.

Nearest Neighbor Search (q, k) // optimal algorithm
1. Initialize ranking = index.increm-ranking (F(q), df)
2. Initialize result = new sorted-list (key, object)
3. Initialize dmax = w
4. While o = ranking.getnext and d,(o, q) I d,,, do
5. If do@, s> s dmax then result.insert (d,(o, q) , o)
6. If result.length 2 k then dmax = result[k].key
7. Remove all entries from result where key > dmax
8. End while Report all entries from result where key I dmax

VI. EXPECTED RESULTS

1. Data Encryption and decryption Result

When RSA algorithm is applied on the data then we get encrypted data. and that encrypted data is store on the cloud. User can access the data after downloading and decrypting file. For encryption and decryption keys are provided

2. Ranking Result

When any User request for the data then Ranking is done on requested data using k-nearest neighbor algorithm. For Ranking co- ordinate matching principle is used. After ranking user gets the expected results of the query.

3. Alert System Results

If any unauthorized User tries to access or updating the data on cloud, then alert will be generated in the form of mail and messages. The alert intimates the authorized user.

VII. CONCLUSION

Thus we proposed the problem of multiple-keyword ranked search over encrypted cloud data, and construct a variety of security requirements. From various multi- keyword concepts, we choose the efficient principle of coordinate matching. We first propose secure inner data computation. Also we achieve effective ranking result using k-nearest neighbour technique. This system is currently work on single cloud Provide better security in multi-user systems.

VIII.REFERENCES

[1] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou,"Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data," Proc. IEEE INFOCOM, pp. 829- 837, Apr, 2011.

[2] E.-J. Goh, ―Secure Indexes,‖ Cryptology ePrint Archive, http:// eprint.iacr.org/2003/216. 2003

[3] N. Cao, C. Wang, M. Li, K. Ren, and W. Lou, ―Privacy-Preserving Multi-Keyword Ranked Search over Encrypted Cloud Data,‖ Proc. IEEE INFOCOM, Jan, 2014.

[4] Towards Secure Multi-Keyword Ranked Search over Encrypted Cloud Data (MRSE)