

# Secure Authorized Deduplication and Differential Query Services in Cost Efficient Cloud

**Sikha Mary Varghese**

*Department of Computer Science & Engineering  
St. Joseph College of Engineering and Technology Palai,  
India*

**Mereen Thomas**

*Department of Computer Science & Engineering  
St. Joseph College of Engineering and Technology Palai,  
India*

## Abstract

The project first makes the attempt to formally address the problem of authorized data deduplication. Here apart from the traditional deduplication systems, the differential privileges of users are further considered in duplicate check besides the data itself. It shows the proposed authorized duplicate check scheme incurs minimal overhead compared to normal operations. The one of the advantage of cloud is that it is cost efficient, so cloud users can tolerate a certain amount of delay in order to reduce cost during information retrieval. The two issues which is addressed here in such an environment is privacy and efficiency. Here two efficient information retrieval for ranked query (EIRQ) schemes to reduce querying overhead is introduced. In EIRQ users can retrieve files on demand by choosing queries of different ranks. It is useful when a large number of matched files are there and the user only needs a subset of them.

**Keywords:** Deduplication, differential privileges, efficient information retrieval for ranked query (EIRQ), matched files

## I. INTRODUCTION

Cloud computing is the long dreamed vision of computing as a utility, where users can remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. By data outsourcing, users can be relieved from the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the possibly large size of outsourced data makes the data integrity protection in Cloud Computing a very challenging and potentially formidable task, especially for users with constrained computing resources and capabilities.

Cloud computing provides many virtualized resources to users as services across the entire Internet, while hiding platform and implementation details. Nowadays cloud service providers offer both highly available storage and massively parallel computing resources at relatively low costs. GMAIL is one of the best examples of cloud storage which is used by most of us regularly. Cloud computing provides a low cost, scalable, location independent infrastructure for data management and storage. The rapid adoption of Cloud services is accompanied by increasing volumes of data stored at remote servers; hence techniques for saving disk space and network bandwidth are needed.

A central up and coming concept in this context is deduplication, where the server stores a single copy of each file, in spite of how many clients asked to store that file. All clients that store the file merely use links to the single copy of the file stored at the server. Moreover, if the server already has a copy of the file then clients do not even need to upload it again to the server, thus saving bandwidth as well as storage. In a typical storage system with deduplication, a client first sends to the server only a hash of the file and the server checks if that hash value already exists in its database. If the hash is not in the database then the server asks for the entire file. Otherwise, since the file already exists at the server it tells the client that there is no need to send the file itself. Both way the server marks the client as an owner of that file, and from that point on there is no difference between the client and the original party who has uploaded the file. The client can therefore ask to restore the file, regardless of whether he was asked to upload the file or not.

Data deduplication is data compression technique for eliminating duplicate copies of repeating data in storage. This technique is used to improve storage utilization and can also be applied to network data transfers to decrease the number of bytes that must be sent. Deduplication eliminates redundant data by keeping only one physical copy and referring other redundant data to that copy instead of keeping multiple data copies with the same content. Deduplication can take place at the file level and the block level. It eliminates duplicate copies of the same file at file level and also eliminates duplicate blocks of data that occur in non-identical files at the block level.

Deduplication lowers storage costs as fewer disks are needed. It improves disaster recovery since there's far less data to transfer. Backup/archive data usually includes a lot of duplicate data. The similar data is stored over and over again, consuming unwanted storage space on disk or tape, electricity to power and cool the disk/tape drives and bandwidth for replication. This will create a chain of cost and resource inefficiencies within the organization.

While providing data confidentiality, traditional encryption is incompatible with data deduplication. Specifically, it requires different users to encrypt their data with their own keys. Thus, indistinguishable data copies of different users will lead to different cipher texts, making deduplication unfeasible. Convergent encryption has been proposed to impose data confidentiality

while making deduplication feasible. It encrypts and decrypts a data copy with a convergent key, which is obtained by computing the cryptographic hash value of the content of the data copy. After key generation and data encryption, users preserve the keys and send the cipher text to the cloud. Because the encryption operation is deterministic and is derived from the data content, indistinguishable data copies will generate the same convergent key and hence the same cipher text.

To avoid unauthorized access, a secure PoW (proof of ownership protocol) is also needed to provide the confirmation that the user indeed owns the same file when a duplicate is found. After the confirmation, consequent users with the same file will be provided a pointer from the server without needing to upload the similar file. A user can download the encrypted file with the pointer from the server, which can only be decrypted by the equivalent data owners with their convergent keys. Thus, convergent encryption will allow the cloud to do deduplication on the cipher texts and the proof of ownership (PoW) prevents the unauthorized user to access the file.

A cost efficient cloud environment can tolerate a certain degree of delay while retrieving information from the cloud to reduce costs. The two fundamental issues in such an environment is privacy and efficiency. First a private keyword-based file retrieval scheme that was originally proposed by Ostrovsky is introduced. Ostrovsky scheme shows very high computational outlay, because the cloud needs to process keywords in each and every file in the cloud. The user can send a query every time that process the query.

Because of this process the cloud is overwhelmed with queries from many users of different organizations. Through this process the communication and computation cost is beyond expectation their scheme allows a user to retrieve files of interest from an untrusted server without leaking any information. The main drawback is that it will cause a heavy querying overhead incurred on the cloud, and thus goes against the original intention of cost efficiency. A scheme, termed efficient information retrieval for ranked query (EIRQ), based on an aggregation and distribution layer (ADL), to reduce querying overhead incurred on the cloud. In EIRQ, queries are classified into multiple ranks, where a higher ranked query can retrieve a higher percentage of matched files. A user can retrieve files on demand by choosing queries of different ranks. This feature is useful when there are a large number of matched files, but the user only needs a small subset of them.

## II. SECURE AUTHORIZED DEDUPLICATION AND DIFFERENTIAL QUERY SERVICES IN COST EFFICIENT CLOUD

For better confidentiality and security in cloud computing a new deduplication construction supporting authorized duplicate check in hybrid cloud architecture, in which the duplicate-check tokens of files are generated by the private cloud server with private keys. Proposed system includes proof of data owner so it will help to implement better security issues in cloud computing.

In a cloud computing environment, an organization subscribes the cloud services and gives access to its staff to share files in the cloud. Each file is identified with some keywords, and the staff, only authorized users can retrieve files, so they send query with those keywords to cloud and retrieve interested files. In such a scenario, protection of user privacy from the cloud, which is outside the security boundary of the organization, this becomes a key problem.

User privacy can be classified into search privacy and access privacy. A naive solution is used to protect user privacy when the files are stored in the clear forms, so that the cloud cannot know which files the user is really interested in. User queries are classified into multiple ranks, so a new kind of user privacy that is rank privacy is introduced in cloud computing. Rank privacy is used to hide the rank of each user query from the cloud. While this does provide the necessary privacy, the communication cost is high. EIRQ protocol is the latest protocols and it addresses the issues of privacy, aggregation, CPU consumption and network bandwidth usage.

In the proposed system, the main aim is to enhance the system security. Specifically, an advanced scheme to support stronger security by encrypting the file with differential privileges is presented. In this way the users without corresponding privileges cannot perform the duplicate check. Furthermore, such unauthorized users cannot decrypt the ciphertext and even collude with the S-CSP.

In most organizations the storage systems contains duplicate copies of many pieces of data. For example, the same file may be saved in several different places by different users. Here two or more files may not be identical still may include much of the same data. Deduplication will eliminate these extra copies by saving just one copy of the data and replacing the other copies with pointers that lead back to the original copy. Companies frequently use deduplication in backup and disaster recovery applications, but it can be used to free up space in primary storage as well.

Here a hybrid cloud is used which is an integrated cloud service. It utilizes both the private cloud and public cloud to perform distinct functions within the same organizations. All cloud computing services should offer certain efficiencies to differing degrees but public cloud services are likely to be more cost efficient and scalable than private clouds. Therefore an organization can maximize their efficiencies by employing public cloud services for all non-sensitive operations and only relying on a private cloud where they require it and ensuring that all of their platforms are seamlessly integrated.

In this deduplication system that supports the hybrid cloud architecture, the Storage –Cloud Service Provider(S-CSP) lies in the public cloud. The user needs to get the file token from the private cloud server to perform duplicate check for some file. The private cloud server also checks the user's identity before issuing the corresponding file token to the user. The authorized duplicate check for this file can be performed by the user with the public cloud before uploading this file. Based on the results of duplicate check, the user either uploads this file or runs Proof of Ownership (PoW).

Apart from this the two issues which is addressed in a cost-efficient environment is the privacy and efficiency. Earlier there was a private keyword-based file retrieval scheme proposed by Ostrovosky, this scheme allows the users to retrieve files of interest from the untrusted server without leaking information, but the disadvantage was it causes a heavy querying overhead which goes against the intention of cost efficiency. So Efficient Information retrieval for Ranked Query (EIRQ) scheme is introduced which reduces the querying overhead.

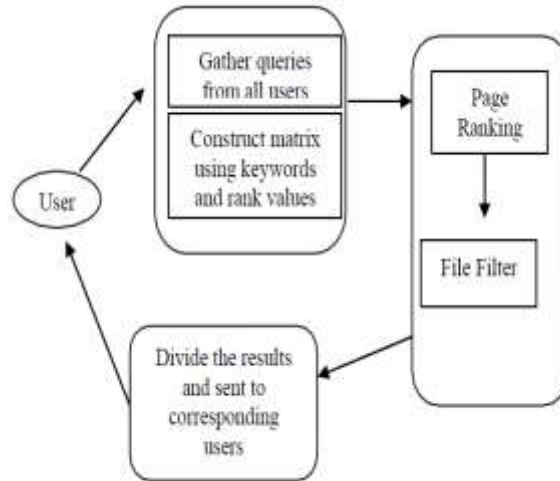


Fig. 1: EIRQ-Efficient Scheme description

Here the authorized users will send the queries to the Aggregation and Distribution Layer (ADL). The ADL aggregate the user's queries and send it as a combined query to the cloud. Then the combined queries are processed by the cloud on the file collection and will send a buffer. The buffer involve of all matched files to the ADL to wait for a period of time before running, which may get a certain querying delay.

The EIRQ scheme is having a user and a cloud. The users are only authorized from the cloud and then only accessing is possible otherwise it is not possible. This process is going on both wired network and wireless network. First it will send a request from the user to ADL for establishment of a connection from the ADL. Then authorized user should have their own login name and passwords. After login, the user will generate a query. This query is encrypted into 0's and 1's and then will send to the ADL. At the ADL Side Matrix Construct algorithm has been done based on keywords and ranks. This process is called aggregation. After the aggregation process, ADL sends the Mask Matrix to the cloud. At the cloud side File Filer algorithm has been done. This algorithm will filter out files based on ranks and keywords.

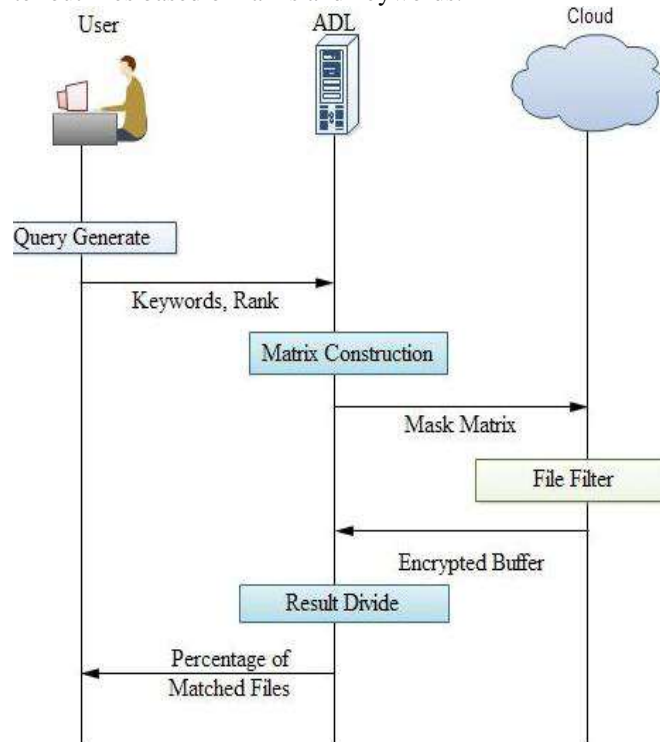


Fig. 2: Working of EIRQ Scheme

The modules split-up are as follows:

- 1) User registration: In this module, user is having authentication and security to access the detail which is in the public cloud. Before accessing or searching the details, user should have an account in that otherwise they should register first.
- 2) Secure deduplication system: the authorized user will be able to use his/her individual private keys to produce query for particular file and privileges he/she owned with the aid of private cloud, while the public cloud performs duplicate check directly and tells the user if there is any duplicate.
- 3) Security of duplicate check token : Here the unauthorized users with inappropriate privileges or file should be prevented from getting or generating the file tokens for duplicate check of any file stored in S-CSP. Also any user without querying the private cloud server for certain file token cannot get any valuable information from the token which includes the file information or privilege information.
- 4) Query generation: The user sends the keywords and the rank of the query to the ADL by using Query Generation algorithm.
- 5) Matrix construction: The ADL runs the Matrix Construct algorithm after aggregating enough user queries, to send a mask matrix to the cloud. The mask matrix  $M$  consists that  $d$ -row and  $r$ -column matrix, where  $d$  is the number of keywords, and  $r$  is the lowest query rank.
- 6) File filter: The cloud runs the File Filter algorithm to return a buffer. The buffer contains a certain percentage of matched files to the ADL. Here the AES algorithm used.
- 7) Result and divide: To distribute search results to each user by the ADL runs the Result Divide algorithm. It requires the cloud to attach keywords to the file content to allow the ADL to distribute files correctly. By executing keyword searches the ADL can find out all of the files that match users' queries.

### III. RESULTS AND DISCUSSIONS

EIRQ schemes can provide search privacy, access privacy, and rank privacy as follows.

#### A. Search Privacy

In the two schemes, the combined query sent to the cloud. is encrypted under the ADL's public key with the Paillier cryptosystem. The query is a matrix of encrypted 0s and 1s. The Paillier cryptosystem is semantically secure, and the cipher text of every 1 or 0 is different from other 1s or 0s. Therefore, the cloud cannot deduce what each user is searching for from the encrypted query.

#### B. Access Privacy

In the two schemes, the cloud processes the encrypted query on each file in a collection, and maps the processing result into a buffer, which is encrypted with the ADL's public key. The cloud conducts this process for all files in the same way. Therefore, the cloud cannot know which files are actually returned from the encrypted buffer.

#### C. Rank Privacy

In EIRQ-Simple, the messages from the ADL to the cloud are  $r$  encrypted queries, the buffer size, and the mapping times, where  $r$  is the information, which we leak more than. Given  $r$ , the cloud only knows the number of query ranks without knowing how many users are in each rank, nor which users are in which ranks. Therefore, EIRQ Simple can protect the basic level of rank privacy for a user. In EIRQ-Efficient, the message from the ADL to the cloud is a  $d$ -row and  $r$  column mask matrix, where  $d$  is the number of keywords in the dictionary, and  $r$  is the lowest rank of user queries. Here,  $r$  is the information that leak. Therefore, EIRQ-Efficient can protect the basic level of rank privacy for a user.

#### D. Computational Cost

Here only consider the cost of the exponential operation, which is the most expensive. In both parameter settings, the results are the same. In EIRQ-Simple, the computational cost is  $r$  times more than No Rank since, for each ranked query, the cloud needs to process it on the file collection once. In EIRQ-Efficient, the computational cost is much the same as in No Rank, since the cloud needs to execute exponentiation once for each file.

As described already, the computational cost is mainly determined by the number of exponentiations performed by the cloud, which is almost the same under the Bloom filter and the Ostrovsky parameter settings. In order to justify the analyses, we will compare the computational cost between keywords and two EIRQ schemes, EIRQ Simple and EIRQ Efficient. The comparisons of computational cost on the cloud are shown in Graph 1, where the number of queries in each rank ranges from 1 to 25. Here queries are categorized under the ranks into 0 to 3 ranks. Rank 0, Rank 1, Rank 2 and Rank 3 should recover the files 100%, 76%, 52%, 24% of matched files. Here the two schemes of EIRQ provide differential query services and no bandwidth wasted for every transaction. Out of this two the EIRQ Efficient proves to be a better solution. Here the two EIRQ schemes are compared on the basis of its computation cost incurred on the cloud.

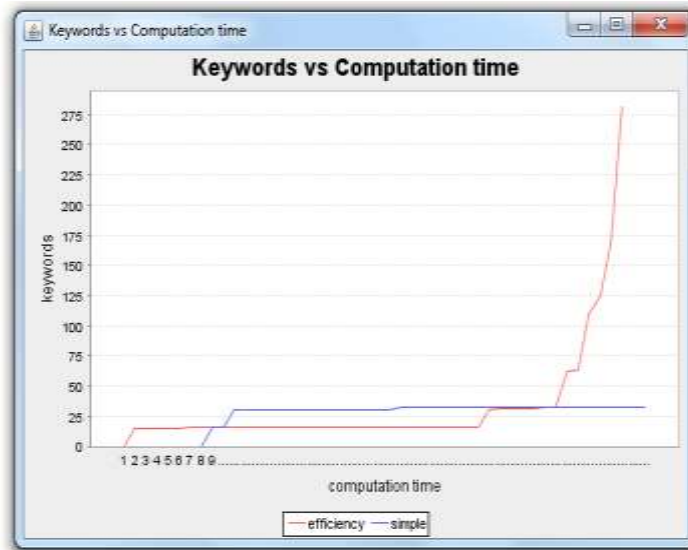


Fig. 3: Graph 1: graph of keywords vs computation time

There are following differences in the Ostrovsky scheme, COPS protocol and EIRQ scheme with respect to various parameters such as security, computational and communicational cost.

Table – 1  
Comparison of different schemes

Protocol	Parameter		
	Security and privacy	Computational Cost	Bandwidth Cost
Otrovsky	Yes	No	No
COPS protocol	Yes	Yes, to some extent	Yes, to some extent
EIRQ	Yes	Yes	Yes

Here also out of the three schemes compared EIRQ scheme is the better scheme satisfying all the parameters. EIRQ- Efficient scheme resolves the two fundamental problems. it will determine the relationship between query rank and the matched files to be returned. The queries are classified into 0 to r ranks. Rank 0 queries have the highest rank and the rank r queries have the lowest rank. Secondly it can determine which matched files will be returned and which will not.

#### IV. CONCLUSION

The authorized data deduplication is proposed to protect the data security by including differential privileges of the users in the duplicate check. The duplicate-check tokens of files are generated by the private cloud server with private keys. The authorized duplicate check scheme incurs minimal overhead compared to convergent encryption and network transfer. The two EIRQ schemes are proposed (EIRQ Simple and EIRQ Efficient) are worked through ADL. It offers differential query services, which will also protect the user privacy. These schemes are provide, clients are recovered certain percentage of matched records by particular queries of various ranks. In this, EIRQ scheme assign ranks for each query, then highest rank files are matched and user recovered certain percentage of matched files.

#### REFERENCES

- [1] M. Bellare, S. Keelveedhi, and T. Ristenpart, "DupLESS: Server-Aided Encryption for Deduplicated Storage", In USENIX Security Symposium, 2013
- [2] P. Anderson and L. Zhang., "Fast and Secure Laptop Backups with Encrypted De-duplication", In Proc. of USENIX LISA, 2010.
- [3] S. Halevi, D. Hamik, B. Pinkas, A. Shulman-Peleg, "Proof of Ownership in Remote Storage System" In 18th ACM Conference on Computer and Communications Security (ACM CCS).
- [4] Jia Xu, Ee-Chien Chang, Jianying Zhou., "Weak Leakage-Resilient Client-side Deduplication of Encrypted Data in Cloud Storage" ,In Institute for Info Comm Research, Singapore, 2013
- [5] Jin Li, Xiaofeng Chen, Mingqiang Li, Jingwei Li, Patric P.C.Lee and Wenjing Lou, "Secure Deduplication with Efficient and Reliable Convergent Key Management " ,In IEEE Transactions on Parallel and Distributed Systems , Vol. 25 , No. 6 , June 2014

- [6] P. Paillier. "Public-Key Cryptosystems Based on Composite Degree Residuosity Classes", in Proc. EUROCRYPT, 1999, pp. 223-238.
- [7] Cong Wang, Ning Cao, Kui Ren, and Wenjing Lou, "Enabling Secure and Efficient Ranked Keyword Search over Outsourced Cloud Data", IEEE Transactions on Parallel and Distributed systems, vol. 23, no. 8.
- [8] Ning Cao, Cong Wang, Li, Ming, Kui Ren, Wenjing Lou, "Privacy-Preserving Multi-keyword Ranked Search over Encrypted Cloud Data", INFOCOM, 2011 Proceedings IEEE April 2011
- [9] R. Ostrovsky and W. Skeith III, "Private searching on streaming data " in Proc. of ACM CRYPTO, 2005.
- [10] Q. Liu, C. Tan, J. Wu, and G. Wang, J., "Cooperative Private Searching in Clouds", Parallel Distrib. Comput. , vol. 72, no. 8, pp. 1019-1031, Aug. 2012.
- [11] M. W. Storer, K. Greenan, D. D. E. Long, and E. L. Miller. Securedata deduplication. In Proc. of StorageSS, 2008.
- [12] M. Mitzenmacher, "Compressed Bloom Filters," IEEE/ACM Trans. Netw., vol. 10, no. 5, pp. 604-612, Oct. 2002.