

# SECURE DATA GROUP SHARING AND CONDITIONAL DISSEMINATION WITH MULTI OWNER IN CLOUD COMPUTING

S. Parthiban<sup>1</sup>, Sharon<sup>2</sup>

<sup>1</sup>*Dept of Computer Science, School of Arts and Science, Vinayaka Mission 's Research Foundation, Chennai, India.  
Email : [sparthi333@gmail.com](mailto:sparthi333@gmail.com)*

<sup>2</sup>*Lecturer, Dept of Computer Science, School of Arts and Science, Vinayaka Mission 's Research Foundation,  
Chennai, India.  
Email: [sharon.csa@avit.ac.in](mailto:sharon.csa@avit.ac.in)*

## Abstract

As a basic query function, range query has been exploited in many scenarios such as Sql retrieves, location-based services, and computational geometry. Meanwhile, with explosive growth of data volume, users are increasingly inclining to store data on the cloud for saving local storage and computational cost. However, a long-standing problem is that the user's data may be completely revealed to the cloud server because it has full data access right. To cope with this problem, a frequently-used method is to encrypt raw data before outsourcing them, but the availability and operability of data will be reduced significantly. In this paper, we propose an Efficient and Geometric Range Query scheme (EGRQ) supporting searching and data access control over encrypted spatial data. We employ secure KNN computation, polynomial fitting technique and order-preserving encryption to achieve secure, efficient and accurate geometric range query over cloud data. Then, we propose a novel spatial data access control strategy to refine user's rights in our EGRQ. To improve the efficiency, R-tree is adopted to reduce the searching space and matching times in whole search process. Finally, we theoretically prove the security of our proposed scheme in terms of confidentiality of spatial data,

privacy protection of index and trapdoor, and the unlink ability of trapdoors. In addition, extensive experiments demonstrate the high efficiency of our proposed model compared with existing schemes.

## 1. INTRODUCTION

Under big data-driven society, data deduplication technique has been widely developed in cloud storage because it can significantly reduce storage costs by storing only a single copy of redundant data. Indeed, data deduplication can reduce storage costs by more than 50% in standard file systems and by more than 90% for backup applications, and these savings translate into substantial financial savings to cloud service providers and users. However, considering security and privacy concerns of outsourced data, users are likely to encrypt data with their own keys before outsourcing. This impedes the cross-user deduplication since an identical data will be encrypted into different cipher texts by different users' keys, i.e., it is challenging to identify duplicates over different cipher texts. Thus, how to efficiently conduct data deduplication over encrypted data becomes a pressing issue.

Data integrity is one of the most important properties when a user outsources its files to cloud storage. Users should be convinced that the files

stored in the server are not tampered. Traditional techniques for protecting data integrity, such as message authentication codes (MACs) and digital signatures, require users to download all of the files from the cloud server for verification, which incurs a heavy communication cost. These techniques are not suitable for cloud storage services where users may check the integrity frequently, such as every hour. Thus, researchers introduced Proof of Storage (PoS) for checking the integrity without downloading files from the cloud server. Furthermore, users may also require several dynamic operations, such as modification, insertion, and deletion, to update their files, while maintaining the capability of PoS.

The project has covered almost all the requirements. Further requirements and improvements can easily be done since the coding is mainly structured or modular in nature. Improvements can be appended by changing the existing modules or adding new modules. One important development that can be added to the project in future is check even the data in system with encryption of the data stored in the file and also we can even modify the security system over the data recover by the user.

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective.

We proposed the comprehensive requirements in multi-user cloud storage systems and introduced the model of deduplicatable dynamic PoS. We designed a novel tool called HAT which is an efficient authenticated structure. Based on HAT, we proposed the first practical deduplicatable dynamic PoS scheme called

DeyPoS and proved its security in the random oracle model. The theoretical and experimental results show that our Dey-PoS implementation is efficient, especially when the file size and the number of the challenged blocks are large.

As described in, secure cross-domain (or cross-user) deduplication schemes are evaluated in terms of two factors: security properties that each scheme provides and system overheads incurred on the cloud service provider and users. However, as far as we know, existing related schemes cannot efficiently achieve the strong confidentiality of outsourced data while resisting brute force attacks. Thus, in this paper, under the similar two-level multi-domain architecture, we propose an efficient and privacy-preserving multi-domain big data deduplication scheme in cloud storage. The main contributions of this paper are three aspects:

- First, we propose a secure multi-domain deduplication scheme that can support data deduplication not only in the same domain (called intra-deduplication) but also across multiple different domains (called inter-deduplication). Specifically, the proposed scheme generates the random convergent key and the random tag based on the bilinear pairing technique to ensure data confidentiality. The proposed scheme only needs to produce a constant number of random ciphertexts to ensure that the outsourced encrypted data can be correctly decrypted by users with ownership. Moreover, the proposed scheme achieves that only the cloud service provider can perform the inter deduplication by comparing random inter-tags derived from the Boneh-Goh-Nissim cryptosystem.
- Second, to improve the time complexity of duplicate search, we construct a deduplication decision tree based on the B+ tree, which works well for the big data storage system. In particular,

the length of plaintext can be used to represent the keyword in the B+ tree.

- Third, we analyze the security of the proposed scheme and demonstrate that it can achieve strong data confidentiality and data integrity while resisting brute-force attacks. Besides, extensive performance evaluations justify the efficiency of the proposed scheme in terms of computational, communication and storage overheads.

## 2. LITERATURE SURVEY

**Title:** Deduplication on encrypted big data in cloud

**Author:** Z. Yan, W. Ding, X. Yu, H. Zhu, and R. H. Deng

**Abstract:** With the continuous and exponential increase of the number of users and the size of their data, data Deduplication becomes more and more a necessity for cloud storage providers. By storing a unique copy of duplicate data, cloud providers greatly reduce their storage and data transfer costs. The advantages of Deduplication unfortunately come with a high cost in terms of new security and privacy challenges. We propose Clouded up, a secure and efficient storage service which assures block-level Deduplication and data confidentiality at the same time. Although based on convergent encryption, Clouded up remains secure thanks to the definition of a component that implements an additional encryption operation and control mechanism

The main advantage of these techniques is that users can share data even when they are offline. The main disadvantage of these techniques is that optimization of design is necessary so that CSP functions properly in Deduplication management.

**Title:** Secure and efficient cloud data Deduplication with ownership management

**Author:** S. Jiang, T. Jiang, and L. Wang

**Abstract:** Data Deduplication has been widely used in cloud storage to reduce storage space and communication overhead by eliminating redundant data and storing only one copy for them. In order to achieve secure data Deduplication, the convergent encryption scheme and many of its variants are proposed. However, most of these schemes do not consider or cannot address the efficiently dynamic ownership changes and the secure Proof-of-Ownership (PoW), simultaneously. In this paper, we propose a secure data Deduplication scheme with efficient PoW process for dynamic ownership management. Specially, our scheme supports both cross-user file-level and inside-user block-level data Deduplication. During the file-level Deduplication, we construct a new PoW scheme to ensure the tag consistency and achieve the mutual ownership verification. Moreover, we design a lazy update strategy to achieve efficient ownership management. For inside-user block-level Deduplication, the user-aided key is used to realize convergent key management and reduce the key storage space.

The main advantage of this technique is that it establishes trust relation among cloud storage components with policy-based Deduplication. The main disadvantage of this technique is that data deletion and owner management is not considered by policy-based Deduplication.

**Title:** Enhanced secure threshold data Deduplication scheme for cloud storage

**Author:** J. Stanek and L. Kencl

**Abstract:** As more corporate and private users outsource their data to cloud storage, recent data breach incidents make end-to-end encryption increasingly desirable. Unfortunately, semantically secure encryption renders various cost-effective storage optimization techniques, such as data

Deduplication, ineffective. On this ground Stanek et al. [1] introduced the concept of “data popularity” arguing that data known/owned by many users do not require as strong protection as unpopular data; based on this, Stanek et al. presented an encryption scheme, where the initially semantically secure cipher text of a file is transparently downgraded to a convergent ciphertext that allows for Deduplication as soon as the file becomes popular. In this paper we propose an enhanced version of the original scheme.

The main advantage of these techniques is that brute force attacks are avoided and clients can encrypt their data with key server which is different from separate storage server. The main drawback of these techniques is that flexibility to other data users can not be provided.

### 3. EXISTING SYSTEM

While most of the searchable encryption schemes focus on common SQL queries, such as keyword queries and Boolean queries, few studies have specifically investigated geometric range search over encrypted spatial data. Wang et al. proposed a novel scheme to specifically perform circular range queries on encrypted data by leveraging a set of concentric circles. Some previous searchable encryptions handling order comparisons can essentially manage axis parallel rectangular range search on encrypted spatial data. Similarly, Order-Preserving Encryption, which has weaker privacy guarantee than searchable encryption, is also able to perform axis-parallel rectangular range search with trivial extensions. Ghinita and Rughin is particularly leveraged certain Functional Encryption with hierarchical encoding to efficiently operate axis-parallel rectangular range search on encrypted spatial data in the application of mobile users monitoring.

### 3.1 DISADVANTAGES OF EXISTING SYSTEM:

- Most of the searchable encryption schemes focus on common SQL queries, such as keyword queries and Boolean queries, few studies have specifically investigated geometric range search over encrypted spatial data.
- Inevitably introduces obstacles in terms of search functionalities over encrypted data.
- None of these previous works have particularly studied geometric range queries which are expressed as non-axis-parallel rectangles or triangles.
- More importantly, there still lacks a general approach, which can flexibly and securely support different types of geometric range queries over encrypted spatial data regardless of their specific geometric shapes.

### 4. PROPOSED SYSTEM:

In this paper, we propose a symmetric-key probabilistic Geometric Range Searchable Encryption. With our scheme, a semi-honest (i.e., honest-but-curious) cloud server can verify whether a point is inside a geometric range over encrypted spatial datasets. Informally, except learning the necessary Boolean search result (i.e., inside or outside) of a geometric range search, the semi-honest cloud server is not able to reveal any private information about data or queries.

Our main contributions are summarized as follows:

- We present a symmetric-key probabilistic Geometric Range Searchable Encryption, and formally define and prove its security with distinguish ability under Selective Chosen-Plaintext Attacks (IND-SCPA).
- In addition, our search process is non-interactive on encrypted data. In terms of search complexity, our baseline scheme incurs linear complexity (with regard to the number of data records), and its

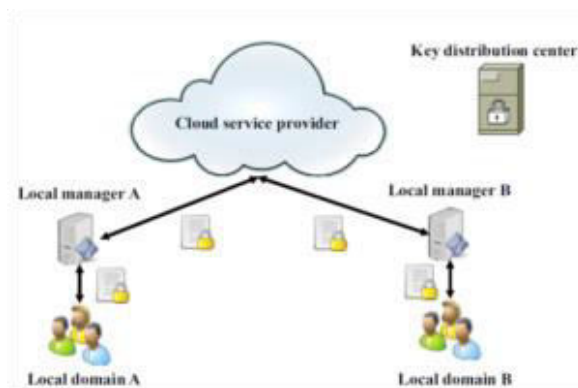
advanced version realizes faster than- linear search by integrating with tree structures.

- Our design is a general approach, which can securely support different types of geometric range queries on encrypted spatial data regardless of their geometric shapes. Furthermore, our design is not only suitable for geometric range queries, but also compatible with other regular types of geometric queries, such as intersection queries and point enclosure queries, over encrypted spatial data.

#### 4.1 ADVANTAGES OF PROPOSED SYSTEM:

- The security of our scheme is formally defined and analyzed with indistinguishability under Selective Chosen-Plaintext Attacks.
- Our design has great potential to be used and implemented in wide applications, such as Location-Based Services and spatial databases, where the use of sensitive spatial data with a requirement of strong privacy guarantee is needed.

#### 5. SYSTEM ARCHITECTURE:



#### 6. MODULE DESCRIPTION

- ✓ Cloud storage process
- ✓ Deduplication Decision Trees
- ✓ Privacy Analysis Encryption Mode.
- ✓ Efficiency Data share

##### 6.1 CLOUD STORAGE PROCES

Every client is affiliated with a domain (e.g., employees in the company or students and faculty members in the university or university network, say University of Texas system). Clients upload and save their data with the CSP. In order to protect their data privacy and help the CSP to complete data deduplication over encrypted data, they encrypt the data and generate the corresponding tags. Finally, clients send message tuples containing encrypted data and the corresponding tags to the LMA or LMB (clients from domains A and B send message tuples to the LMA and LMB, respectively). Clients are considered honest. In theory, it is possible that they would collude with the CSP to obtain other clients' privacy. As mentioned in [14], in practice, such collusion may result in significant risks to the reputation of the CSP, as well as civil litigation or criminal investigations. In addition, if the CSP colludes with client A to compromise the privacy of client B, the CSP is also likely to collude with client B or other clients to compromise the privacy of other existing clients. This would have serious repercussions for the CSP if such collusion is reported or known. Thus, we assume that the CSP does not collude with its clients. Other than brute-force attacks, we do not consider other active attacks.

##### 6.2 DEDUPLICATION DECISION TREES

The process of finding the duplicated data is equivalent to the lookup operation, and the process of adding a new data is equivalent to the insertion operation. In addition, if the client needs to delete a specific data stored in the CSP, then the CSP can use the deletion operation of BST, while ensuring the balance of DDTs. In order to improve the efficiency of finding duplicated data, we construct the deduplication decision trees (DDTs) based on the popular binary search tree (BST) for searching duplicate data. As far as we know, the

DDT initialization is similar to insertion of the BST, but this operation begins with the empty tree. the current moment. Hence, CSP constructs a DDT-A for this domain to store  $k$  message tuples for subsequence deduplication. Based on the insertion operation of the BST, we propose Algorithm 1 to construct DDTs. According to Algorithm 1, CSP stores  $k$  message tuples in turn at appropriate nodes, as shown in Fig.2. In addition, in order to ensure the Time complexity of searching duplicate is  $O(\log k)$ , we need to balance the tree in the process of the DDT construction.

### 6.3 PRIVACY ANALYSIS ENCRYPTION MODE

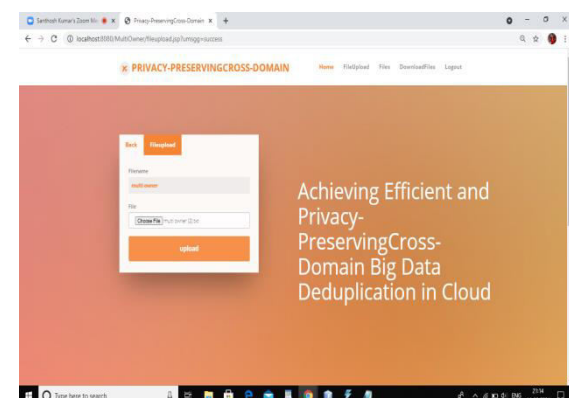
We analyze that our EPCDD scheme can protect the privacy of sensitive data from disclosure, and minimize the duplicate information disclosure. In order to cooperate with the CSP to process data deduplication, clients need to not only upload the encrypted data, but also provide two corresponding tags. Because  $C_i$  is encrypted by the symmetric encryption algorithm, i.e., AESCBC, the security of  $C_i$  is based on the symmetric encryption algorithm. Moreover, if the CSP and LMA (LMB) want to obtain  $m_i$  from the tag  $1_i = g^{h_2(m_i)}$  ( $1_i = g^{h_2(m_i)}$ ), it means that they need to deal with the discrete logarithm (DL) problem and the one-way hash function, which have been proved to be computationally infeasible difficult problems. Therefore, CSP or LMA (LMB) cannot obtain  $m_i$  from the  $1_i$ . Because  $2_i = s_{ki} \bmod !$ , there is an integer  $k$  such that  $s_{ki} = k! + 2_i$ . One equation has two unknown numbers, CSP or LMA (LMB) only can obtain  $s_{ki}$  by guessing attack. However, as described in section 4.2, if  $j_{skij} = 256$  bits, we can set  $j_{lj} = 128$  bits, which can sufficiently resist the guessing attack. Therefore, data confidentiality can be achieved in this paper. In addition, to verify whether the different ciphertexts correspond to the

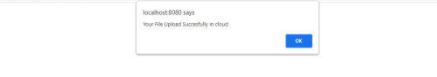
identical plaintext, it needs to verify whether the Eq. (1) holds. As designed in our EPCDD scheme, only the CSP has the secret parameters  $g_{aq}$  and  $g_{bq}$ , so that only it can perform this verification. With secret key  $s$  and parameter  $e(g; g)^t$ , the client computes the message-dependent symmetric key  $s_{ki} = h_1(\text{mike}(g; g)^{st})$ . Then, this client chooses a random number  $r_i \in \mathbb{Z}_N$ , computes the ciphertext  $C_i = \text{Enc}_{s_{ki}}(r_i \parallel m_i)$ , where the symmetric encryption algorithm is the cipher block chaining (CBC).

### 6.4 EFFICIENCY DATA SHARING


Privacy. Although the CSP and local managers can obtain the encrypted data along with the corresponding tags, the CSP and LMA (LMB) are not able to obtain plaintexts from these tuples. In addition, the duplicate information disclosure is inevitable in data deduplication, but we seek to minimize such information leakage as much as practical availability. In order to reduce the big data storage and management overheads, the CSP may attempt to delete the duplicate data, without affecting data availability Efficiency. The storage, computation and communication overheads associated with the big data deduplication should be as small as possible, and the cost of searching for duplicated data should also be minimized.

## 7. RESULTS





The screenshot shows a web browser window with the address bar displaying 'localhost:8080'. A security warning dialog box is open, stating 'localhost:8080 says Your file loaded insecurely in about:'. The dialog box has an 'OK' button. The browser's address bar shows 'localhost:8080' and the page title is 'localhost:8080 says'. The browser's address bar also shows 'localhost:8080' and the page title is 'localhost:8080 says'.

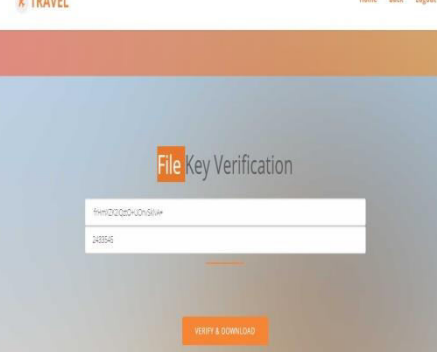
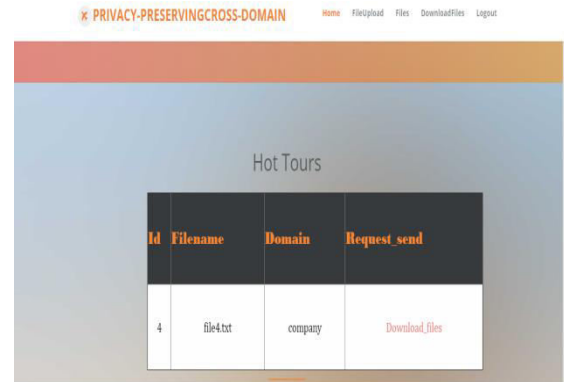


cloud File Details

Id	Filename	Domain	Request send
3	file4.txt	company	File Request send
4	file4.txt	company	File Request send
5	file5.txt	company	File Request send
6	newtxt.txt	company	File Request send
7	multi owner.txt	company	File Request send
8	owner.txt	company	File Request send
9	multi owner.txt	company	File Request send
10	new.txt	company	File Request send
11	new.txt	company	File Request send
12	new.txt	company	File Request send
13	save.txt	company	File Request send
14	copy.txt	company	File Request send
15	part.txt	company	File Request send
16	part.txt	company	File Request send
21	multi owner.txt	company	File Request send
22	multi owner (2).txt	company	File Request send

File Request From User

Id	Filename	Domain	Request_send
9	multi_owen.txt	company	File Request_send
10	new.txt	company	File Request_send
11	new.txt	company	File Request_send
12	new.txt	company	File Request_send
14	copy.txt	company	File Request_send
15	parth.txt	company	File Request_send
17	banu.txt	students	File Request_send
18	banu1.txt	students	File Request_send



TRAVEL

Home Back Logout

## File Key Verification

9fcm2Q2G0uOn8iUw

Verify

VERIFY & DOWNLOAD

X TRAVEL

The data security and privacy is a concern for users in cloud computing. In particular, how to enforce



privacy concerns of multiple owners and protect the data confidentiality becomes a challenge. In this paper, we present a secure data group sharing and conditional dissemination scheme with multi-owner in cloud computing. In our scheme, the data owner could encrypt her or his private data and share it with a group of data accessors at one time in a convenient way based on IBBE technique. Meanwhile, the data owner can specify fine-grained access policy to the ciphertext based on attribute-based CPRE, thus the ciphertext can only be re-encrypted by data disseminator whose attributes satisfy the access policy in the ciphertext. We further present a multiparty access control mechanism over the ciphertext, which allows the data co-owners to append their access policies to the ciphertext.

## REFERENCES

- [1] Z. Yan, X. Li, M. Wang, and A. V. Vasilakos, "Flexible data access control based on trust and reputation in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 5, no. 3, pp. 485-498, 2017.
- [2] B. Lang, J. Wang, and Y. Liu, "Achieving flexible and self-contained data protection in cloud computing," *IEEE Access*, vol. 5, pp. 1510-1523, 2017.
- [3] Q. Zhang, L. T. Yang, and Z. Chen, "Privacy preserving deep computation model on cloud for big data feature learning," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1351-1362, 2016.
- [4] H. Cui, X. Yi, and S. Nepal, "Achieving scalable access control over encrypted data for edge computing networks," *IEEE Access*, vol. 6, pp. 30049-30059, 2018.
- [5] K. Xue, W. Chen, W. Li, J. Hong, and P. Hong, "Combining data owner-side and cloud-side access control for encrypted cloud storage," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 8, pp. 2062-2074, 2018.
- [6] C. Delerablée, "Identity-based broadcast encryption with constant size ciphertexts and private keys," *Proc. International Conf. on the Theory and Application of Cryptology and Information Security (ASIACRYPT '2007)*, pp. 200-215, 2007.
- [7] N. Paladi, C. Gehrman, and A. Michalas, "Providing user security guarantees in public infrastructure clouds," *IEEE Transactions on Cloud Computing*, vol. 5, no. 3, pp. 405-419, 2017.
- [8] J. Bethencourt, A. Sahai, and B. Waters, "Ciphertext-policy attribute based encryption," *Proc. IEEE Symposium on Security and Privacy (SP '07)*, pp. 321-334, 2007.
- [9] L. Liu, Y. Zhang, and X. Li, "KeyD: secure key-deduplication with identity-based broadcast encryption," *IEEE Transactions on Cloud Computing*, 2018, <https://ieeexplore.ieee.org/document/8458136>.
- [10] Q. Huang, Y. Yang, and J. Fu, "Secure data group sharing and dissemination with attribute and time conditions in Public Clouds," *IEEE Transactions on Services Computing*, 2018, <https://ieeexplore.ieee.org/document/8395392>.
- [11] Box, "Understanding collaborator permission levels", <https://community.box.com/t5/Collaborate-By-Inviting-Others/UnderstandingCollaborator-Permission-Levels/ta-p/144>.
- [12] Microsoft OneDrive, "Document collaboration and co-authoring", <https://support.office.com/en-us/article/document-collaborationand-co-authoring-ee1509b4-1f6e-401e-b04a-782d26f564a4>.



[13] H. He, R. Li, X. Dong, and Z. Zhang, "Secure, efficient and finegrained data access control mechanism for P2P storage cloud," IEEE Transactions on Cloud Computing, vol. 2, no. 4, pp. 471-484, 2014.

[14] Z. Qin, H. Xiong, S. Wu, and J. Batamuliza, "A survey of proxy reencryption for secure data sharing in cloud computing," IEEE Transactions on Services Computing, 2018, <https://ieeexplore.ieee.org/document/7448446>.

[15] J. Son, D. Kim, R. Hussain, and H. Oh, "Conditional proxy reencryption for secure big data group sharing in cloud environment," Proc. of 2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 541-546, 2014