

## PERFORMANCE EVALUATION OF VARIOUS DATA DEDUPLICATION SCHEMES IN CLOUD STORAGE

B.Tirapathi Reddy<sup>1</sup>, Dr.M.V.P.Chandra Sekhara Rao<sup>2</sup>

<sup>1</sup>Associate Professor, K.L.University, India

<sup>1</sup>tirapathireddy@kluniversity.in

<sup>2</sup>Professor, RVR&JC College of Engineering, India

**Abstract:** With the increased interest in collecting data, the need for data storage is increasing, which in turn increases the cost of backup and maintenance. Conventional data storage and backup mechanisms will not ensure Elimination of redundant data and incur a lot of management overhead. If redundant information was stored in backup devices, it wastes a lot of storage space and consumes lot of computational power for accessing and updating of data. And with the wide popularity of the cloud computing data users are attracted to the cloud storage and using cloud storage as a backup for storing valuable information. But uploading the data to remote cloud storage raises security concerns about the confidentiality and integrity of the data. So the need for secure cloud storage, which eliminates redundant data, maintains confidentiality and integrity of data are also increasing. Data Deduplication is a mechanism most popularly used to eliminate redundant copies from the cloud storage. This paper compares various data deduplication mechanisms available, list out various merits and demerits

**Keywords:** cloud storage, convergent encryption, data deduplication, block level deduplication, file-level deduplication

### 1. Introduction

Cloud Storage: In the present scenario of the technological world, each and every user handles a lot of data and information. Each user has data for his/her personal as well as professional usage. But the internal storage capacity of each user PC doesn't support much to this cause. Many researchers tried to find a solution for this storage problem as a result finally termed a solution where any user can store his unlimited data and information in a centralized distributed storage location which is called as a

Cloud. A cloud is a combination of many services: Platform-as-a-service, Infrastructure-as-a-service and Software-as-a-service. Cloud provides a user with the required hardware and software tools that is required for him/her or any application that the user wants over the internet. This is the done using platform-as-a-service. Infrastructure-as-a-service is a cloud computing model where a third-party host's various set of virtualized computing resources over the internet. Software-as-a-service is a software licensing model where the trending and legacy software's are hosted at a central location and is licensed on a subscription basis. These are accessed most often via a thin client namely web browser. As a lot of user's data has to be outsourced to the cloud, there is a huge requirement of transmission bandwidth and storage capacity. These two aren't a big concern in the present scenario but in the years to come the data will be increased by many folds, as a result, the transmission bandwidth and the storage capacity are going to be a huge concern.

Cloud Computing is a term based on utilities and consumption of virtualized resources. It provides deployment models and services to the cloud. It allows many kinds of information resources to be uploaded for existing process to render computing reports without any need to store data in the cloud. Cost-efficient method to use, maintain and upgrade and it has unlimited space to store data on the cloud and has automatic software integration, very easy to access the data, last and important is we have quick deployment even though it takes the time to complete. Cloud is used as pay-as-you-go. Cloud computing is a new environment that deploys on infrastructure deployment which supports on-demand services and software. Cloud storage refers to store the information online in the cloud. It offers you to store manage and access data very easily on cloud's infrastructure. Data can be stored in digital,

physical and logical pools with multiple spans of servers. Cloud storage improves the application's performance to keep costs low. Storage has greater elasticity and scalable to backup data, no need to purchase any storage from the cloud. [2] We can access it for free to store our data in the cloud. But most of the data stored in the cloud is redundant, i.e. several users are uploading the same copy of the data. Thereby the storage space gets wasted. So in order to utilize the storage space effectively and to eliminate redundant copies of the data, data deduplication has been the most widely used mechanism.

## 2. Data Deduplication

Data deduplication mechanism is a process of automatically removing redundant data from the storage/ backup systems. Data deduplication compares the data items with other data items (files/blocks) and deletes the redundant data items. Data deduplication can be done at the user side i.e. prior to uploading the data item to the cloud storage or can be done on the server side, i.e. after the data has been uploaded to the remote storage server. Deduplication can be done at several levels of granularity i.e. file level or block level. In file level deduplication entire file be compared with the files of the storage system to verify whether the same file exists or not. Whereas in block level deduplication, the file will be divided into several blocks, each block will be compared, with the blocks stored in the storage server to verify whether the block is redundant or not. Over a period of time client side/user side deduplication mechanism has been the widely used technique, because it saves uploading costs, network bandwidth. Both file level and block level deduplication mechanisms have their own advantage and disadvantages.[3][4] if we implement file level deduplication, a small change in the file causes the entire file to be saved again, which violates the purpose of deduplication technique. Block level deduplication, in this case, would only save the changed block, instead of the entire file[8]. Thereby exhibits better performance; on the other hand, the indexes generated to identify the file are smaller in file level deduplication when compared to block level deduplication. But there are many challenges to be addressed when we are implementing data deduplication.

When we are uploading the data to a remote storage like cloud storage, preserving the confidentiality of the data is also important [1]. But

data deduplication is not compatible with the conventional encryption mechanisms that we use to convert the data into a secret form. In conventional encryption, the user first will encrypt the data with a key, that he chooses, and sends the ciphertext to the cloud storage, but if several users try to upload the data using the conventional encryption mechanism, there will be so many different cipher texts uploaded to the cloud storage. so cloud storage provider will not able to identify whether two copies uploaded belonging to the same copy of the plain text or not. So if users are interested in making the data confidential, they have to sacrifice storage efficiency. if storage efficiency is to be given importance, users have to sacrifice on confidentiality of the data.

So to ensure confidentiality of the data, while avoiding the redundancy of the data, i.e. maintaining the storage efficiency, a mechanism called convergent key encryption has been introduced. Convergent key encryption is a process of encrypting the data with a key generated by applying the data to a hash function; the resulting hash code will be used as the key to encrypt the data. The user will keep the hash code called convergent key with him and uploads the resulting ciphertext to the cloud storage.

Cloud storage provider will maintain the copies of the cipher text and a tag derived from the ciphertext. The user who is trying to upload a file to the cloud storage will send the tag to the cloud storage provider. Cloud storage service provider will compare the tag with the tags available on the server. If there is a tag match then it is assumed that same copy of the file is already available on the cloud storage server and the user need not upload the file to the cloud storage server. The user will be added to the list of owners for that particular file. That is how convergent key encryption helps us in implementing data deduplication process.

## 3. Challenges in data deduplication systems

Convergent encryption is providing a mechanism for us to achieve both confidentiality and deduplication facility. But convergent key encryption has its own limitations. convergent key encryption is suffering from brute-force attack . Especially with the block level Convergent key encryption, the number of hash codes generated will be too large, because each user will be associated with a convergent key for a block and that too several users may upload the same block. so the number of keys to be managed will become large in no time. So managing that much number of keys will be an

overhead task. If convergent keys are to be stored in cloud storage, it wastes so much of storage space also. So managing a large number of keys generated is a challenging task. Some work has already been done in this regard.

Another challenge is, in deduplication systems to verify for the deduplication of data a small hash code will be used as a proxy. The cloud storage provider permits anyone who produces the hash code to access the corresponding file. Suppose A is a user uploading a file X, while trying to upload he stores the hash value of the file X in the cloud storage. If User A want to share this file With another User B, He will simply send the hash code of file X to user B. Meanwhile if User C gets that hash value, he can cheat the cloud storage provider as follows, User C might upload a file Y to the cloud Storage provider, but instead of storing the hash code belonging to the file Y, he will store the hash code of the file X. Later C Sign Up to the same cloud storage server asks to backup the file Y, but the hash code of the file x to the cloud storage provider instead of the hash code of the file Y. Later C asks to download the file by submitting the Hash code of the file, now Cloud service provider thinks that C owns the File X and gives him the control to access the file X, though C does not own the file. So using the small hash code as a proxy to access the file has created so many security Problems, to solve this problem, proof of ownership is introduced. .proof of ownership is all about convincing the cloud service provider that, the file we are trying to upload is actually owned by us. to verify the proof of ownership, the cloud service provider will ask us to perform certain operations on the cipher text, the same will be performed by the cloud service provider also. if the results of us are matched with the results of the cloud service provider, proof of ownership is verified.

Bellare et al [5] developed a Server aided encryption scheme for deduplication Storage. DupLESS is a server side deduplication mechanism in which clients encrypts the files using message based keys derived from an oblivious PRF protocol in this method data deduplication happens on the server side. the storage requirements are bit high in DupLESS when compared to the other state-of-the-art mechanisms available. Dupless is quite easy to deploy in the cloud environment. it overcomes the brute force attack by maintaining a key server which can manage the keys securely. if the key is the hashes of the messages, brute force attack is quite

possible. so to avoid that a separate Key server will be maintained, which will generate and manage the keys independently.

armknecht et al [12] proposes a transparent data deduplication scheme named ClearBox by which users can verify the effective storage space they are using. they can pay the amount for which they are using only. it ensures that the data files are accessible to only genuine owners only . it allows the users to have fine-grained control over the files . earlier mechanisms help the cloud storage servers to realize the benefit of data deduplication techniques. but clear box will help the user to realize the benefit of data deduplication mechanism.the author has deployed this in Amazon S3 and DropBox and evaluated the overhead and performance benefits to the cloud service user.

Roberto Di Pietro et al[11]proposed a mechanism to provide proof of ownership with data deduplication. Whenever a new file is uploaded the server checks whether the file is already present in the database or not. If the file is not there then it computes new challenges and it stores those queries and answers for those queries in the database and the file is uploaded and a hash value will be given. The hashing is done using hashmap data structure. Now, when the client requests for the download of the file, server poses a challenge which is not yet used. At this instant client has to prove his ownership by giving the correct responses. If one fails to give correct responses then he will not be allowed to access that particular file. but the biggest drawback of this mechanism is it wastes so much of storage space, the computational complexity is high for the user using this mechanism. it is vulnerable to various attacks because the challenges are stored in an array, if anyone gets that array he/she can able access the contents easily.

jin li et al[7] proposed a mechanism called dekey which manages the huge number of convergent keys that are produced in client-side block level data deduplication schemes. reduces the storage space and bandwidth costs a lot, provides confidentiality of the data .it applies the deduplication techniques to the convergent keys used for encrypting the data. it leverages the benefits of secret key sharing techniques. the secret key is split into n number of subkeys and are distributed to several key servers which are independent. the first user who tries to upload the file will calculate the key and split them into n shares, will distribute them to key servers. to access the file the user has to

access at least  $k$  servers, prove his authentication, will able to recompute the key and can access the file. this will reduce the storage overhead of the convergent keys, and will make the key management efficient and reliable. to securely distribute the convergent keys among the key servers Ramp-Secret key sharing is used.

M. Bellare et al [6] formalized a cryptographic primitive called message locked encryption, in which the key used for encryption is derived from the message itself. by nature confidentiality and data deduplication is not compatible. to make data deduplication compatible with ensuring confidentiality, deriving the key from the message itself is proved to be the right choice. the mechanism by which the deduplication is made compatible with conventional encryption is called convergent key encryption, also called message locked encryption. the main advantage with message locked encryption is it provides stronger security for unpredictable messages. but semantically message locked encryption does not provide stronger security to messages, which are predictable in nature.

Jin Li et al . Proposed a hybrid cloud approach for secure authorized deduplication [10] scheme for addressing the problem of authorized data deduplication . in this users will be given different privileges and these will be considered to check for data deduplication of data items. every user is given some access privileges to access the files. the access rights to the file are based on the set of privileges. in order to upload the file , the data owner, the data owner has to interact with the private cloud, has to prove his identity, with the private key . after verifying the identity of the user, private cloud server will check the corresponding privileges assigned to the user. the user will calculate the tag and sends the file tags to the public cloud storage service provider. if a file duplicate is found at the cloud storage service provider, the user needs to run a Proof of ownership Protocol, to prove that he actually own that particular file. then cloud service provider verifies whether the user owns the copy or not. if owner proves the ownership he will be made the owner of the file. the main advantage of this mechanism is the user can access the file only if he/she possess the privileges to access. the hybrid cloud approach consisting of private cloud to verify the privileges and ownership are effective and secure because the data items are stored in the public cloud storage server, which the user can access only after getting the permission from a Private cloud server.

Waraporn Leesakul et al[14] proposed dynamic data deduplication scheme in which the author tried to balance data deduplication and quality of service. as too much of data is to be stored , and it should be available to the users all the time. unlike the other static data deduplication mechanisms which will consider all data access scenarios uniformly, this mechanism tried to bring balance between data deduplication and fault tolerance. because data access rates change over a period of time. some data items may be accessed most frequently in a particular period of time, but may not be required in the other interval of time. so there must be a compromise between the level of deduplication and fault tolerance. in general, the hash value of the file will be used to verify whether the same copy of the file is existing in the cloud storage or not. some mechanisms will statically make use of the number of references created for that file as an indicator, to decide whether the file can be replicated or not. To improve data availability, while ensuring effective use of storage space, considered dynamic access of data items and quality of service of a cloud environment. the number of replicas of the file will be changed dynamically based on the number of references made to the file.

Jan Stanek et al [15] has proposed a secure data deduplication scheme for cloud storage. due to various weaknesses in convergent key encryption and client side deduplication is leaking so much of information to unauthorized users. there are different types of data items stored in the cloud storage, which requires varying levels of protection. depending on the number of instance of the data items stored in the cloud storage classified the data items as popular and unpopular. popular data items require less security as they are owned by many users and are not sensitive. whereas unpopular data items require stronger security as they are owned by very few people and are sensitive in nature. introduced a threshold cryptosystem, that will help us to classify whether the item stored is popular or unpopular . an identity manager, verifies the identity of the users. indexing service will help to have a secure indirection of the unpopular data items.

#### 4. Conclusion

In this paper, we have evaluated merits and demerits of various data deduplication mechanisms, and the practical difficulties in implementing those mechanisms. as users are interested in collecting a lot of data, the importance of effective data

deduplication mechanisms is increasing day by day. various challenges have been identified in the process of making data deduplication practical to the cloud storage user. especially the need for ensuring the privacy of data stored in cloud storage is improving day by day . all the authors used mostly convergent encryption to make data deduplication practical, but convergent has its own limitations .our work proceeds in developing an efficient data deduplication mechanism, the preserves the confidentiality of data and privacy of the data owners, with efficient key management.

### References

- [1] B.Tirapathi Reddy, U.Ramya, Dr.M.V.P.Chandra Sekhara Rao, "A comparative study on data deduplication techniques in cloud storage", IJPT| Sep-2016 | Vol. 8 | Issue No.3 | 18521-18530.
- [2] B.Tirapathi Reddy, M.V.P Chandra Sekhara Rao, Dr.L.S.S Reddy "A Survey on Assured File Deletion in Cloud Environment", International Journal of Applied Engineering Research ISSN 0973-4562 Volume 9, Number 23 (2014) pp. 19899-19907
- [3] Xu, Jia, and Jianying Zhou. "Leakage-resilient proofs of ownership in cloud storage, revisited." International Conference on Applied Cryptography and Network Security. Springer International Publishing, 2014.
- [4] Halevi, Shai, et al. "Proofs of ownership in remote storage systems." Proceedings of the 18th ACM conference on Computer and communications security. ACM, 2011.
- [5] M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless: Server aided encryption for deduplicated storage. In USENIX Security Symposium, 2013
- [6] M. Bellare, S. Keelveedhi, T. Ristenpart. Message locked encryption and secure deduplication. In EUROCRYPT, pages 296–312, 2013.
- [7] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou. Secure deduplication with efficient and reliable convergent key management. In IEEE Transactions on Parallel and Distributed Systems 2013.
- [8] C.Ashwini, J.Muthu, B.Karthikeyan, V.Vinith Raj, "An Efficient Auditing Technique for Secure Cloud Computing Using Asymmetric Cryptographic Algorithm", International Innovative Research Journal of Engineering and Technology, volume 1, no.1 (2015), pp. 23-26.
- [9] P. Anderson and L. Zhang. Fast and secure laptop backups with encrypted de-duplication. USENIX LISA, 2010
- [10] J. Xu, E.-C. Chang, and J. Zhou. In ASIACCS, pages 195–206, 2013. Weak leakage resilient client side deduplication of encrypted data in cloud storage.
- [11] Li, Jin, et al. "A hybrid cloud approach for secure authorized deduplication." IEEE Transactions on Parallel and Distributed Systems 26.5 (2015): 1206-1216
- [12] Di Pietro, Roberto, and Alessandro Sorniotti. "Proof of ownership for deduplication systems: a secure, scalable, and efficient solution." Computer Communications 82 (2016): 71-82.

