Special Issue on Proceedings of International Conference on Newer Trends and Innovation in Nanotechnology, Materials Science, Science and Technology March 2021. International Journal of Aquatic Science, Vol 12, Issue

CLOUD STORAGE ENVIRONMENT IN A SECURE CLIENT SIDE DEDUPLICATION

M. Jebakumari, M.E, K. Arun Patrick,

Assiatant Professor& Professor, Department of CSE, Nehru Institute of Technology, Coimbatore,

Tamil Nadu, India

Correspondent author: nitarunpatrick@nehrucolleges.com

Abstract — **Data mining brings forth lot of information extraction security and privacy concerns for user's sensitive data and it ensures security to both inside and outside attacks in cloud storage. Traditional encryption, while providing data confidentiality, is incompatible with data Deduplication. So for more advanced security it requires a stronger encryption system. Traditional encryption allows different users to encrypt their data with their own keys. Thus, identical data of different users leads to attaining different cipher texts; this affects deduplication of identical data. Convergent encryption has been suggested in order to enforce data confidentiality for making Deduplication feasible. It encrypts or decrypts a data copy with a convergent key, A cryptographic hash value is generated by computing the contents of the data, which are uploaded. After generating the key and encryption of data, users gain the keys and send back the cipher text to the storage. Since the encryption operation is deterministic and obtained by comparing the actual data, identical data copies will generate the same convergent key and hence the same Ciphertext. Data Deduplication has been a conventional method in maintaining validity in the files and to remove same data files. It helps to reduce the amount of storage space and save bandwidth during upload operation. To ensure the confidentiality of sensitive data while supporting Deduplication, the convergent encryption technique has been suggested to encrypt the data before outsourcing or deploying. The objective is for making the better environment for data security and makes the attempt to formally address the problem of authorized data Deduplication in Cloud Storage**.

Keywords* — *Deduplication; Convergent Encryption; Data mining; Cipher Text

I. INTRODUCTION

Cloud computing allows access to resources from anywhere and at any time through the internet. From the customer's point of view, the customers can reduce their expenditure in purchasing and maintaining storage infrastructure while only paying for the required amount of storage, which can be scaled-up and down upon demand. This assistance makes cloud system more advantageous than other storage systems. But it is also very true that cloud Storage is not infinite. In order to compromise with this problem, data deduplication is introduced. Data deduplication is the best way to handle these unwanted excess data.

Cloud Computing:

Cloud Computing is an internet based computing scheme which supports shared computer processing and provides storage as a service to a heterogeneous community of users. The name cloud comes from the metaphor of word Internet and from complex infrastructure which contains in-system diagrams. The Figure1 represents the outline of devices connected to the cloud network. Cloud computing entrusts services with a user's data, software, and computation over a network.

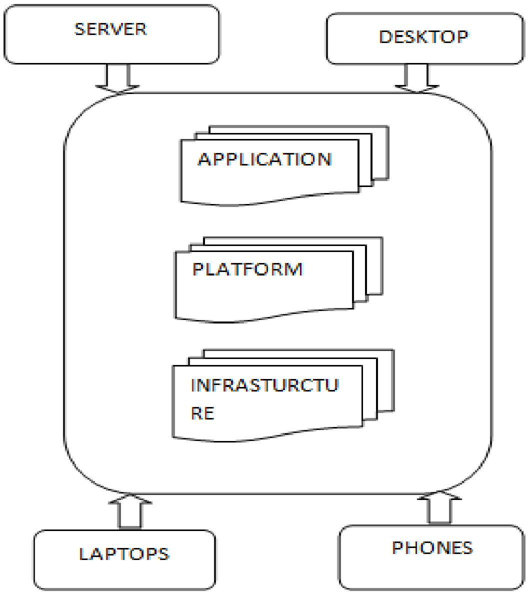

Fig 1 Cloud Computing

II. SYSTEM DESIGN AND ARCHITECTURE

The Fig 2 illustrates an admin controlling rights to different users. The client obtains permission from the admin using a generated OTP transferred to the mail client. File is uploaded and duplicate files are checked. The file is uploaded if and only if another file of the same content is not present in the cloud. When the file being uploaded is unique, a 16 bit unique key is encrypted using MD5 hashing algorithm. The file is then stored in the cloud and can be accessed by the admin and the client.

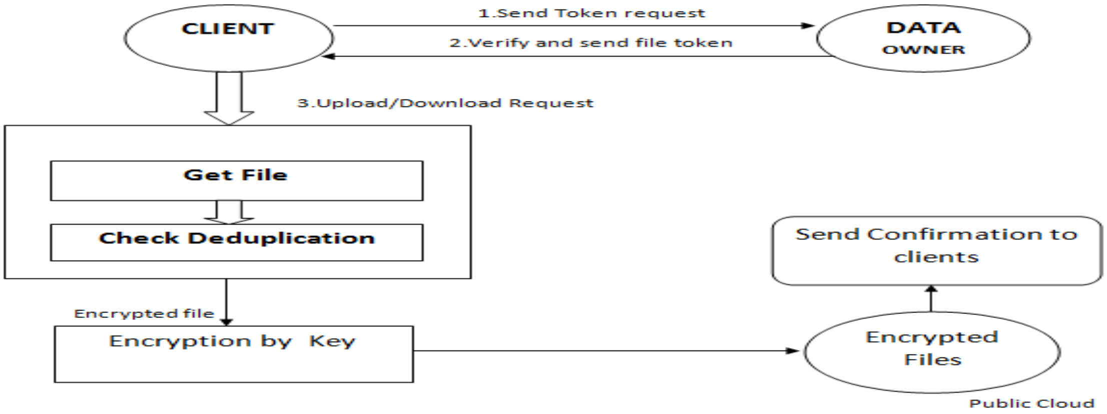

Fig 2 System Design and Architecture

III. EXISTING SYSTEM

The convergent encryption technique has been implemented to encrypt the data before outsourcing for better protection and data security. The previous attempts shows us the problem of authorized data Deduplication with consideration to the file name rather than the attributes of the file. Thus we present several new Deduplication constructions supporting authorized duplicate check. Each deduplication techniques has been developed with different designs considering the characteristics of data sets, System capacity and deduplication type. If datasets have similar files rather than identical file, deduplication should look inside the files to check what parts of the contents are the same as previously same data for better storage space savings. If a system removes redundancies directly in a right path even in a confined storage space, it is better to eliminate redundant data before storage. On the other hand, if a system has residual time or enough space to store data temporarily, deduplication can be performed after the data are placed in temporary storage. Deduplication is classified based on granularity. The unit of compared data can be at the File level of Sub File level, which are further sub divided into fixed size blocks, variable size chunks, packet payload or byte streams in packet payload. The smaller the granularity used, the large number of indexes created, but the more redundant data is detected and removed.

Disadvantages of the existing system:

- The traditional convergent encryption scheme has always been insecure for predictable files.
- There may be a same file name repeated which may conflict.
- The memory usage would increase if at all multiple files having same content and different file names are uploaded

IV. PROPOSED SYSTEM

In this system, the introduction of an advanced deduplication system supporting authorized duplicate check and that which compares the storage system with the file content is seen. In this new deduplication system, the user shall be restricted from uploading the same hash value of the data because it compares the whole data base storage system, which means that it can prevent the duplication process with the same content. The authorized duplicate check for this file content can be performed by the Nave Bayes classifier on the server storage before uploading a file. We propose an effective approach to verify data ownership and check duplicate storage with security challenges that uses proxy re-encryption data support using AES algorithm.

Block-level and File level are the commonly used deduplication methods. In file level deduplication only a single copy of the duplicated file is stored in the storage. Two or more files are recognized as duplicates if they have the same hash value. Hash based deduplication breaks data into chunks, either of fixed or variable length for block level deduplication. Hashing algorithm is implemented on the chunks or processes to generate hash values. The commonly used algorithm to create a hash against the chunks or files are secure hash algorithm 1 (SHA1) and Message Digest Algorithm 5 (MD5). The hash is generated onto a bit string conventionally 128 bits for MD5 and 160 bits for SHA1 that represents the data process.

Using Tom Cat Server the connections are established with the local host system. The user needs to access the code only after an OTP generation. After receiving the admin permission the user can upload the files to the server or desired storage. While uploading, the file undergoes the comparison of codes and is merged along with a hash value of 16 bit. The filename is stored along with the auto generated hash key in SQL database. When a file is uploaded that has the same name it performs a check on replication of the incoming new data or file. The process gets implemented on the background of the database.

Advantages of the system:

- Its efficiency, effectiveness and applicability.
- Our scheme can flexibly support data update and sharing
- The system is focused on being more secure.
- The main purpose is to protect the data security by including differential privileges of users in the duplicate check.

V. METHODOLOGY

Procedure:

- A registration is required for users to handle the system. For registration it is necessary of the admin to grant permission, an OTP generated by the admin is sent to the given E-mail id of the user. AES Encryption technique is used for user registration. While registering it demands for an unique 16 bit private key .
- After successful registration, the user can upload the data using the help of a web portal.
- The uploaded data undergoes Hashing algorithm and a random key value is generated. Using MD5 algorithm the data is analyzed and a hash value is issued based on the content of the file.
- The next stage involves checking the existence of duplicate data. It is performed by comparing the generated Key value of uploaded data with key values of files that are already store in the system.
- While searching, if the entry is not found then it uploads the data to the storage. And if found undergoes deduplication algorithm in order to avoid the redundant data.
- For every operation an SQL table is maintained. It contains the details of user, admin, and uploaded data etc.

SQL Table details:

Attributes	Data Types	Size	Description
Email	Varchar	20	Email of the user
Permission	Varchar	20	To indicate the permission from admin (Yes/No).
OTP	Varchar	20	Generated OTP for specified user.

Table 1: Admin

Attributes	Data Types	Size	Description
Username	Varchar	20	Name of the user
Password	Varchar	20	Password of the user
Gender	Varchar	20	Gender of the user
Email	Varchar	20	Email id of the user

Mobile	Varchar	10	Mobile number of user
Address	Varchar	30	Address of the user
ProductKey	Varchar	30	16-Bit Key given by the user

Table 2: User Registration

Attributes	**Data Types**	**Size**	**Description**
Username	Varchar	20	Name of the user
Password	Varchar	20	Password of the user
Hashvalues	Varchar	20	Hash values generated using MD5
Filename	Varchar	20	File name of the uploaded data
Rootdetails	Varchar	10	Local Disk address
Rootfilename	Varchar	30	File name of the data in Local Disk
Mail_ID	Varchar	30	Email id of the user

Table 3: Hash Algorithm

Usage of Tables:

Table 1: Admin - To store the details of authorized users.

Table 2: User Registration - To store the details of registered users.

Table 3: Hash Algorithm - To store the MD5 Hash values generated for corresponding file which are uploaded to the Local Disk.

Work flow diagram*:*

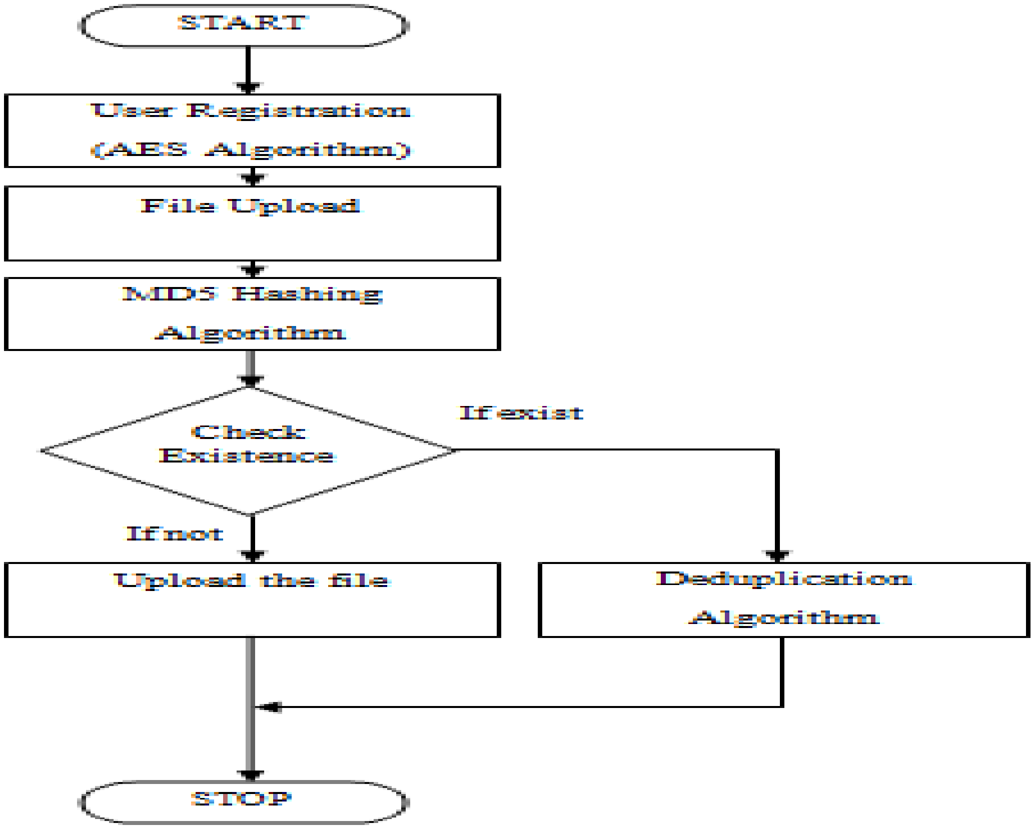

Fig 3 : Working flow diagram

VI. CONCLUSION

Data deduplication is implemented with a specialized data compression technique for eliminating duplicate copies of repeating data. This technique is used to improve storage utilization and can also be applied to network data transfers to more security. Encrypted data can be securely accessed because only authorized data holders can obtain the symmetric keys used for data decryption. On the basis of the code verification and compiling, showed that the proposed system is secure and efficient under the described security model and will be very suitable for big data deduplication. The system is capable for storing data to cloud system, that are designed to avoid replicated data's. The deduplication technique will work on the files with same data but having different file names. This system supports administrator to directly deal with the storage as well as the user stored files. The described system successfully implemented with security system which encrypts the files with a 16-bit key with is given by the user.

REFERENCE

[1] C. Fan, S. Y. Huang, and W. C. Hsu, “Hybrid data deduplication in cloud environment,” in Proc. Int. Conf. Inf. Secur. Intell. Control, 2012, pp. 174–177,doi:10.1109/ISIC.2012.6449734.

[2] C. Y. Liu, X. J. Liu, and L. Wan, “Policy-based deduplication in secure cloud storage,” in Proc. Trustworthy Comput. Serv., 2013, pp. 250–262, doi:10.1007/978-3-642-35795-4_32.

[3] C. Yang, J. Ren, and J. F. Ma, “Provable ownership of file in deduplication cloud storage,” in Proc. IEEE Global Commun. Conf., 2013, pp. 695–700, doi:10.1109/GLOCOM.2013.6831153.

[4] Danny Harnik; OdedMargalit; Dalit Naor; Dmitry Sotnikov; Gil Vernik "Estimation of deduplication ratios in large data sets,"2012 IEEE 28th Symposium on Mass Storage Systems and Technologies (MSST) Year: 2012, DOI: 10.1109/MSST.2012.6232381.

[5] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, “Improved proxy re-encryption schemes with applications to secure distributed storage,” ACM Trans. Inform. Syst. Secur., vol. 9, no. 1,pp.1–30,2006, doi:10.1145/1127345.1127346.

[6] G. Wallace, et al., “Characteristics of backup workloads in production systems,” in Proc. USENIX Conf. File Storage Technol., 2012, pp. 1–16.

.J. Li, Y. K. Li, X. F. Chen, P. P. C. Lee, and W. J. Lou, “A hybrid cloud approach for secure authorized deduplication,” IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 5, pp. 1206–1216, May 2015, doi:10.1109/TPDS.2014.2318320.

[7] J. Paulo and J. Pereira, “A survey and classification of storage deduplication systems,” ACM Comput. Surveys, vol. 47, no. 1, pp. 1–30, 2014, doi:10.1109/HPCC.2014.134.

[8] J. R. Douceur, A. Adya, W. J. Bolosky, D. Simon, and M. Theimer,“Reclaiming space from duplicate files in a serverless distributed file system,” in Proc. IEEE Int. Conf. Distrib. Comput. Syst., 2002, pp. 617–624, doi:10.1109/ICDCS.2002.1022312.

[9] J. W. Yuan and S. C. Yu, “Secure and constant cost public cloud storage auditing with deduplication,” in Proc. IEEE Int. Conf. Communic. Netw. Secur., 2013, pp. 145–153, doi:10.1109/CNS.2013.6682702.

[10] L. J. Gao, “Game theoretic analysis on acceptance of a cloud data access control scheme based on reputation,” M.S. thesis, Xidian University, State Key Lab of ISN, School of Telecommunications Engineering, Xi’an, China, 2015.

[11] M. Bellare, S. Keelveedhi, and T. Ristenpart, “DupLESS: Server aided encryption for deduplicated storage,” in Proc. 22nd USENIX Conf. Secur., 2013, pp. 179–194.

[12] M. Bellare, S. Keelveedhi, and T. Ristenpart, “Message-locked encryption and secure deduplication,” in Proc. Cryptology—EUROCRYPT, 2013, pp. 296–312, doi:10.1007/978-3-642-38348-9_18.

[13] M. Fu, et al., “Accelerating restore and garbage collection in deduplication-based backup systems via exploiting historical information,” in Proc. USENIX Annu. Tech. Conf., 2014, pp. 181–192.

[14] M. Keller; T. Kerins; W. Marnane, "Pairing based cryptography," Second International Conference - Pairing 2008, Egham, UK-September 2008. LNCS 5209.