International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Privacy Preserving and Storage Optimization on Cloud

Prajakta B. Mane[1], N. B. Pokale[2]

[1]Department of Computer Engineering, TSSM Bhivarabai Sawant College of Engineering & Research, Narhe 41, Pune, India

[2]Professor, Department of Computer Engineering, TSSM Bhivarabai Sawant College of Engineering & Research, Narhe 41, Pune, India

Abstract: ***An increasing amount of data is being stored in cloud-based storage services and this trend is expected to grow in the coming years. Data deduplication becomes more and more a necessity for cloud storage providers. Cloud storage services commonly use duplication, which is useful in storing only a unique file or block by eliminating duplicate copies of data. Deduplication is helpful in saving network bandwidth and storage space which is an advantage to the users or clients in cloud. The proposed scheme in this paper not only the reduces the cloud storage capacity but also improves the speed of data deduplication. To protect confidentiality of sensitive data while supporting deduplication the encryption technique has been proposed to encrypt the data before outsourcing.***

Keywords: Cloud Storage, deduplication, Fingerprint, Optimization, Security

1. Introduction

In a cloud storage environment, the storage spaces are provided by third-party organizations. The data from the users are stored in the provided space. The storage space can be provided by the single host, but instead it is provided by multiple sources and it is distributed by the centralized management. The storage protocols that are very often seen are SAN and NAS. Sometimes, the managers of the cloud storage are unable to maintain the efficiency of the storage nodes resulting in increased complexity to control the hardware and network traffic [1][2].

The services provided by cloud computing technique are classified into two major services namely computing and storage. There are various techniques and strategies presented to deal with the challenges related to data storage, data compression and file chunking. The resources that are wasted due to process of revisions, alterations in the data are usually failed to notice. For example, a data file that is re-uploaded may make changes on the network bandwidth, the servers load and also the efficiency. As the cloud network gives a vast scope, the customers are growing widely. And it is difficult to handle the situation when there are multiple users writing their data to the storage and may result in having the similar or identical data. Also because of the vivid habits of the users and the available resources, many users tend to access the similar data and operate the similar functions on the data, thus the system manager cannot guarantee the optimal status of the storage nodes. The cloud storage system networks are enlarging, and increasing the data integration bottleneck and wastage of resources. Even if the system is flexible and rapid, the system may produce duplicate data creating a drawback [3][4]. The term Fingerprint Index Server (FIS) is important for performing the task. It is used to process the functions of the cloud system, including chunk matching, data deduplication, file compression, real-time feedback control, IP information, etc. We propose a technique which consists of INS along with security properties to control and optimize the storage nodes. Due to the proposed work the storage nodes are able to maintain its position and provide proper resource to the clients. Moreover, to balance the load in the system, we use FIS to dynamically monitor IP information and busy level index to avoid network congestion or long waiting times during transmissions [5].

2. Related Work

A. Secure Deduplication

With the advent of cloud computing, secure data deduplication has attracted much attention recently from research community.

Yuan [6] proposed a deduplication system in the cloud storage to reduce the storage size of the tags for integrity check. To enhance the security of deduplication and protect the data confidentiality.

Bellare et al. [3] showed how to protect the data confidentiality by transforming the predictable message into unpredictable message. In their system, another third party called key server is introduced to generate the file tag for duplicate check.

Stanek et al. [4] presented a novel encryption scheme that provides differential security for popular data and unpopular data. For popular data that are not particularly sensitive, the traditional conventional encryption is performed. Another two-layered encryption scheme with stronger security while supporting deduplication is proposed for unpopular data. In this way, they achieved better tradeoff between the efficiency and security of the outsourced data.

Li et al. [4] addressed the key management issue in block-level deduplication by distributing these keys across multiple servers after encrypting the files.

B. Proof of ownership

Halevi et al. [8] proposed the notion of proofs of ownership (PoW) for deduplication systems, such that a client can efficiently prove to the cloud storage server that he/she owns a file without uploading the file itself. Several PoW constructions based on the Merkle-Hash Tree are proposed

[4] to enable client-side deduplication, which include the bounded leakage setting. Pietro and Sorniotti [7] proposed another efficient PoW scheme by choosing the projection of a file onto some randomly selected bit-positions as the file proof. Note that all the above schemes do not consider data privacy. Recently, Ng et al. [7] extended PoW for encrypted files, but they do not address how to minimize the key management overhead.

Q.He et al. [13] talk about various deduplication techniques. The techniques relies on the principle is to maintain only one copy of the duplicate data and a pointer to point to all the duplicate copies. The three types of deduplication can be pos-sible at file level, block level or byte level. The old and new data are compared at byte level and if they match, they are marked as duplicate and pointers are updated.

He et al. [15] discuss various cloud storage techniques. About data deduplication technology, they suggest to retain only the unique instance of the data, reducing data storage volumes. Data deduplication engine creates an index of the digital sig-nature for the data segment and the signature of a given repository to identify data blocks. The index provides a pointer to determine whether the data block is already present. In the copy operation, the data deduplication software found in a block of data inserts a link to the original data block index location instead of storing the data block again. If the same block appears more than once, more pointers to the indexing table are generated. Data migration of cloud storage means moving data from one storage system to another which are at different geographical locations. It aims at cooperating and keeping load balance in cloud storage system. The data should be migrated into other cloud storage units and while keeping pointers in the old stored positions intact, or modify and update the index as changes occur. However it may bring overhead to network bandwidth and access bottleneck to concurrent clients.

3. Methodology

The technique of deduplication over a cloud has been around for few years. The current market products provide excellent deduplication of data for their clients. However, the existing products work in the following manner:

1) If a single file of data has many copies, a single copy is maintained and all the other copies are deleted. However, a pointer pointing to the existing copy indicates how many original copies were present. When more than one copy is required, the pointers point to the existing copy and the user gets the file that he was looking for.
2) While sending these file over the net, all the copies are sent which makes deduplication not very worthwhile and increases the bandwidth requirement.

We propose a system, where in, while sending these files over the network, only fingerprint of the file is sent. On the receiver's side when a user desires for this block, the correct block of data is returned. Thus, reducing the bandwidth and saving the storage costs.

In our propose system uses the Fingerprint index server (FIS) to process cloud storage functions, including file compression, chunk matching, data de-duplication. Therefore, our proposed FIS can manage and optimize the storage nodes according to the client-side transmission conditions so that every storage node can maintain its optimal status and provide suitable resources to clients.

Figure 1 shows the system architecture.

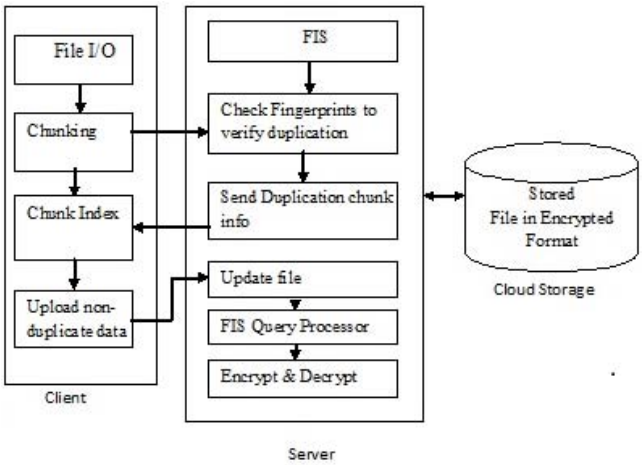

Figure 1: System Architecture

A. Fingerprint Index Server (FIS)

FIS is similar to Domain Name System (DNS) structure, manages the cloud data by a complex P2P-like architecture. Although FIS resembles DNS in structure and functions, FIS mainly processes the one-to-many matches of the storage nodes. IP addresses and hash codes. In general, FIS has following functions:

- Switching between the fingerprints and their corresponding storage nodes.
- Satisfying client demands for transmission as much as possible.

For file transmission optimization, every FIS has exclusive databases of its own domain, which include the fingerprints and their corresponding storage nodes. However, for WAN cloud network environment, to manage the file system by few FIS will cause great burden on the FIS. Therefore, based on the existing DNS structure, we propose to divide the FIS according to the domain FIS and adopt hierarchical management architecture to reduce the workload of the FIS. The FIS mainly query and control the data between fingerprints and storage nodes, and coordinate the transmission by feedback control between storage node and clients. FIS records fingerprints and storage nodes of all data chunks. The FIS record only the locations of the fingerprints and manage the storage nodes.

B.Query Processing in FIS

Every domain-based FIS has databases of fingerprints and storage nodes. The database of fingerprints records the fingerprints of different files and their corresponding storage nodes. When a user looks for specific fingerprints, the FIS queries and confirms if the file already exists in the storage node within the domain before taking the next step. While the clients want to access data, they can use the fingerprints obtained as the index and query the FIS of the upper layer, which searches for the best access node based on the content in the database in case the inefficiency of the access node or

data loss. Different requirements will lead to different query results. If the file that the client wants to access does not exist in the storage nodes in the local domain, the FIS queries the FIS of the upper layer. With the help of the Bloom Filter, the FIS can find out the domain of the FIS with that file chunk and also the accurate storage node through the destination FIS for transmission. Because we consider the workload of the FIS in different layers, the FIS in this article are divided into several layers and have the server client relationships with one another in a hierarchical architecture, that is, the FIS of the upper layer provides service to the FIS of the lower layer only. The burden of each FIS thus can be distributed efficiently.

C. Data Server

The Fingerprints calculated by FIS are stored in database along with fileid, userid and location of file where it is stored on cloud. Data server perform operations Inserting data,Retriving data and deleting data.

D. Cloud Storage

In cloud computing files are stored as per their specific storage space allocated by server.Files stored on cloud with fingerprint in encrypted format for privacy preserving purpose.

E.Set Theory:

Let S, be a system such that,

$\mathbf{S} = \{s, e, X, Y, T, f_{me}, f_{friend}, MEM_{shared}, CPU_{CoreCnt}\}$

Where,

S:Is the Proposed System of Secure storage optimization on cloud.
s: Initial state at T it Performing File Chunking.

e: End state of system

- Store Encrypted File on Cloud space.

X:Input of System.

- It will be Files/Data

Y: Output of System.

- Retrieve the Decrypted file data.

T:Set of serialized steps to be performed secure storage optimization on cloud. -File Chunking, Calculate Fingerprints, Store Fingerprints, Checking Duplication, Store fingerprints and storage location.
f_{me}**:**Main algorithm resulting into outcome Y, mainly focus on success defined for the solution.

- Encryption Algorithm.

MEM_{shared}**:** Memory required processing all these operations, memory will allo-cate to every running process.
$CPU_{CoreCnt}$**:**More the number of count doubles the speed and performance.

F. Algorithm

1) Select File to be upload.
2) Generate a Fingerprint of selected file.
3) Check for deduplication
4) If deduplication then
   upload file in user database give pointer to server database.
   Else
   Store Fingerprint and maintain index of file.
   End if
5) Upload file on cloud
6) Perform Encryption on File using cryptographic algorithm.
   Perform compression on encrypted file.
7) store compressed files on cloud storage.

4. Results

The Table 9.1 shows the result of storage size. Here results compare file size before deduplication and After deduplication.

File Name	Before Deduplication	After Deduplication
TestText1	825	275
TestImage	310	155
Audio file	98	49
TestPdf	15	30

The following Figure 2 graph shows the results of optimized storage after performing deduplication on files. In proposed system we used deduplication techniques. Deduplication reduces the storage size by avoiding same copy of files stored again.

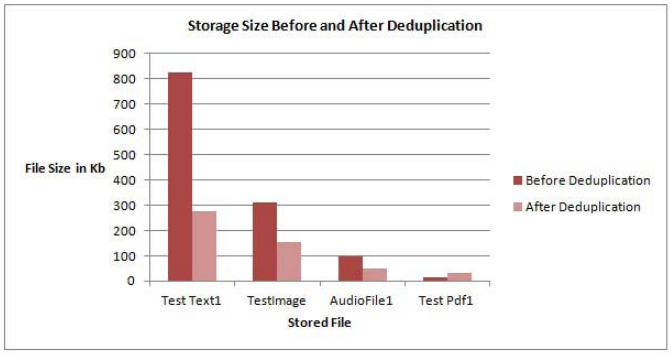

Figure 2: Storage graph showing size of storage

The following Figure 3 graph shows the results of Authorized and Un-Authorized user Access. On Internet No of users access any files that having no security.In our proposed system provided security allows the access of authorised user only.

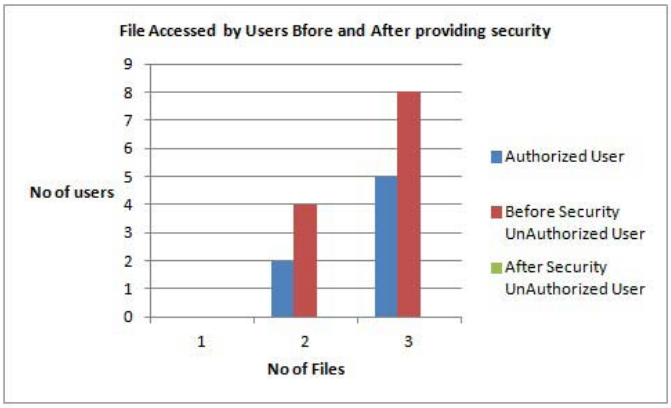

Figure 9.2: Graph For Authorized and UnAuthorized User Access

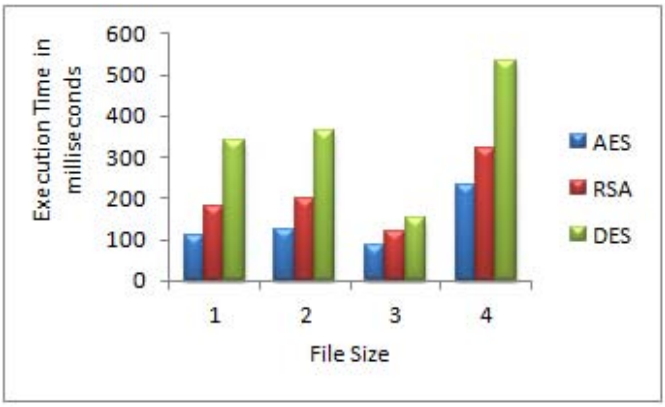

Figure 3: Comparison of encryption algorithm

As we have used AES encryption algorithm in our proposed work, we have shown the comparison with the other encryption schemes. As the file size varies, every encryption algorithm takes different execution time. AES consumes the least execution time as compared to DES and RSA algorithm as shown in figure 3. Using proposed system we reduces the storage space of cloud environment.

5. Conclusions

Finding enough storage space for storing all the data has become a challenging task for many computer owners. As a result many users invest in hard drives. But this is not sufficient. The users then need to delete older folders in order for making space for storage. But nowadays, a smarter way is introduced for storing the data, known as cloud storage. This technique is gaining its popularity in recent years. We have proposed a system that deal with the file handling and security issues related to cloud storage. The Fingerprint Index Server in the proposed work not only works on file compression, chunking, data de-duplication, but also on the processes of encryption, decryption, file storage, optimized node selection, load balancing.

6. Acknowledgement

We thank all the anonymous reviewers and editors for their valuable comments and suggestions to improve the quality of this manuscript.

References

[1] Tin-Yu Wu, Member, IEEE, Jeng-Shyang Pan, Member, IEEE, and Chia-Fan Lin ”Improving Accessing Efficiency of Cloud Storage Using De-Duplication and Feedback Schemes”,IEEE SYSTEMS JOURNAL, VOL. 8, NO. 1, MARCH 2014
[2] J. Yuan and S. Yu. Secure and constant cost public cloud storage auditing with deduplication. IACR Cryptology ePrint Archive, 2013:149, 2013.
[3] M. Bellare, S. Keelveedhi, and T. Ristenpart. Dupless: Serveraided encryp-tion for deduplicated storage. In USENIX Security Symposium, 2013.
[4] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl. A secure data dedu-plication scheme for cloud storage. In Technical Report, 2013.
[5] J. Li, X. Chen, M. Li, J. Li, P. Lee, andW. Lou. Secure deduplication with efficient and reliable convergent key management. In IEEE Transactions on Parallel and Distributed Systems, 2013.
[6] R. D. Pietro and A. Sorniotti. Boosting efficiency and security in proof of ownership for deduplication. In H. Y. Youm and Y. Won, editors, ACM Symposium on Information, Computer and Communications Security, pages 8182. ACM, 2012.
[7] W. K. Ng, Y. Wen, and H. Zhu. Private data deduplication protocols in cloud storage. In S. Ossowski and P. Lecca, editors,Proceedings of the 27th Annual ACM Symposium on Applied Computing, pages 441446. ACM, 2012.
[8] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg. Proofs of owner-ship in remote storage systems. In Y. Chen, G. Danezis, and V. Shmatikov, editors, ACM Conference on Computer and Communications Security, pages 491500. ACM, 2011.
[9] S. Bugiel, S. Nurnberger, A. Sadeghi, and T. Schneider. Twin clouds: An architecture for secure cloud computing. In Workshop on Cryptography and Security in Clouds (WCSC 2011), 2011.
[10] K. Zhang, X. Zhou, Y. Chen, X.Wang, and Y. Ruan. Sedic: ”Privacya ware data intensive computing on hybrid clouds”. In Proceedings of the 18th ACM conference on Computer and communications security, CCS11, pages 515526, New York, NY, USA, 2011. ACM.
[11] G. Urdaneta, G. Pierre, and M. Van Steen, A survey of DHT security tech-niques, ACM Comput. Surveys (CSUR), vol. 43, no. 2,pp. 8:18:49, Jan. 2011.
[12] C.-Y. Chen, K.-D. Chang, and H.-C. Chao, Transaction pattern based anomaly detection algorithm for IP multimedia subsystem”, IEEE Trans. In-form. Forensics Security, vol. 6, no. 1, pp. 152161, Mar. 2011.
[13] Y.-M. Huo, H.-Y. Wang, L.-A. Hu, and H.-G. Yang, A cloud storage ar-chitecture model for data-intensive applications, in Proc. Int. Conf.Comput. Manage., May 2011, pp. 14.
[14] Q. He, Z. Li, and X. Zhang, Data deduplication techniques, in International Conference on Future Information Technology and Management Engineering, 2010, pp. 431432.