A Data Backup Technique in Cloud Computing for Secure Healthcare Data Storage and Disaster Recovery Optimization

Dr. Amina Rahman[1]*
Dr. Victor Chen[1]
Prof. Elena Rossi[1]
[1] University of Singapore, Department of Cloud Computing and Health Informatics Systems, Singapore, Singapore

ABSTRACT

In cloud storage efficiency and data integrity are the two important requirements. This benefits in sparing efforts on heavy data maintenance, management. Single copy of each file is stored in cloud even it is owned by number of users. By reducing reliability deduplication system improves the storage utilization. In this paper we study the problem of integrity auditing and secure deduplication on cloud data and to recovering the files in case of the file deletion or if the cloud gets destroyed due to any reason. To achieve both data integrity and deduplication in cloud, presenting two secure systems SecCloud and SecCloud+. SecCloud introduces an auditing entity with a maintainence of a MapReduce cloud,which helps clients create data tags before uploading as well as audit the integrity of data having been saved or stored in cloud. SecCloud+ is designed that customers always want to encrypt their data before uploading and enables integrity auditing and secure deduplication only on encrypted data. Focuses on the security concept for the back-up files stored at proxy server.

KEYWORDS: SecCloud, SecCloud+, MapReduce cloud, Proof of ownership, Proxy.

1. INTRODUCTION

Cloud computing is a technology that is used for storing and accessing. Cloud storage is a model of computer data storage where in logical pools the digital data is stored.The properties are, they attract more and more customers to use and for storage of their personal data to cloud storage. According to analysis report the volume of data cloud is expected to achieve 40 trillion giga bytes in 2022.Evenly it is widely adopted but fails to accommodate main needs such as the abilities of auditing integrity of cloud files by the cloud clients and detecting duplicated files by the cloud servers. Initial problem is the integrity auditing. Data uploads in different manner like packets, tokens ie less secure because if any of the packet is lost while transmitting it will occur problem for client. It is important that, integrity of the data should be maintained in the storage system. Second problem is secure deduplication in cloud storage. Most of the datas stored at remote storage servers are duplicated. Cloud servers keep only a single copy of each data file and make a link to the file for every client who asks to store the data file. For getting both integrity auditing and secure deduplication in cloud presenting two systems SecCloud and SecCloud+. SecCloud introduces an auditing entity with maintenance of a MapReduce cloud which helps clients to generate datatags before uploading as well as audit the integrity of data having been saved or stored in cloud. SecCloud+ is designed and motivated by the fact that data, data files are encrypted before uploading, integrity auditing and secure deduplication are enabled on the encrypted data.

Drawbacks on existing system

Initially it is very difficult to audit the huge files and the large amount of files using integrity auditing. This also includes lots of duplicate files in the cloud.

Security problems in cloud computing includes on the infected application, data and privacy issues.

To overcome these problems introducing two secure systems SecCloud and SecCloud+ which generates a better and efficient systems for accessing massive data on cloud storage. Initially the plain data file is encrypted and performing the integrity auditing on the encrypted file.

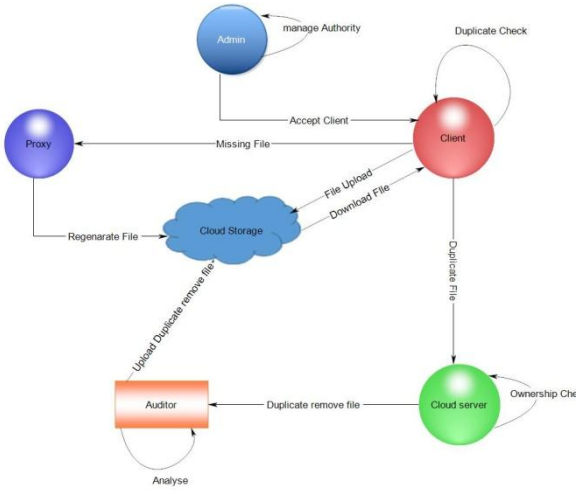
Proposed system

Designing secure deduplication systems having higher reliability in cloud computing. deduplication systems are provided in distributed cloud storage servers to provide better fault tolerance. Secret sharing technique is utilized to protect data confidentiality which is also compatible with the distributed storage systems. Generating an efficient system for accessing massive data on cloud. Encrypting the plain data files and performs integrity auditing on that encrypted files and also focusing on simplicity of the back-up and recovery process.

2. MATERIALS AND METHODS

In this proposed method providing security to the sensitive data that has been stored in cloud and in the proxy server, which has been backup from the cloud storage.

Figure:



In this technique the files can be uploaded and downloaded by the clients. Initially the client's request to upload or download the file must be accepted by the admin. In the proposed system the modules are Admin, Cloud Server, Clients, Auditor, Proxy and Cloud Storage. Cloud clients stores large data files in the cloud. Depends confidently on the cloud for data maintenance and computation. Cloud Clients are either individual consumers or commercial organizations.

While the file uploading in the cloud storage it is encrypted and is stored in cloud by the technique such as Seccloud and Seccloud+. SecCloud includes both integrity auditing and file deduplication on plain files. Here server doesn't know the contain in the file. The functionalities of both integrity auditing and deduplication are applied on plain files.

If there is any duplicated file found, it is passed to the cloud server to provide the ownership. The verification is done in the MapReduce cloud ie having the full details of the files stored in cloud. After that duplicated files are also provided the ownership. After observed by the Cloud server it is passed to the Auditor for analyzing and initial duplicated removed file is passed to the cloud storage. A proxy server is introduced for recovering the files in case of file deletion or the cloud gets destroyed due to any reason.

The files in the cloud storage while backup is performing it is also copied on the proxy server in the same form. The files from the proxy can be directly taken by the admin and the registered clients. Admin and users should have separate key to access the data files. Hence the data should be decrypted and shown to the users. Whenever they want to access their data from the proxy server they should verify by their identity.

While in the Cloud storage the owner of the file should accept the request of other clients to view his/her file.

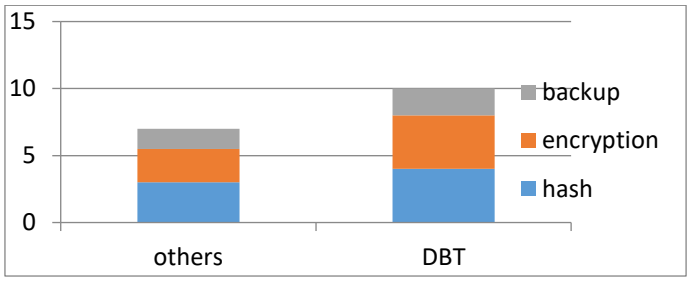
In the proxy server after positive verification the request are provided with attributes. This ensures high degree of encapsulation of the data.

The proxy server should satisfy the issues such as Data integrity, Data confidentiality, Data security, Trustworthiness, Cost efficiency.

While the file is uploading to the cloud, hashing is performed by using SHA 512 algorithm and the hash value is verified with the MapReduce cloud.MapReduce cloud is certain data storage area where all the cloud storage files hash value is kept. And if there is any same hash values while comparing with the MapReduce cloud proof –of –ownership is provided. After that encryption is performed by using Glenc algorithm while storing in cloud.

Proxy is another server which is linked with cloud storage used to perform high security for the files in the cloud storage. In proxy server backup is performed by using MECC Algorithm. Admin and users can directly take files from proxy. Whenever they want to access their data they should verify by their identity.

3. RESULTS AND DISCUSSION



Comparison of the proposed DBT technique and other conventional schemes. The proposed DBT technique is faster than the conventional techniques above graph. Therefore it is clear the DBT technique is efficient than other existing schemes.

4. CONCLUSION

Examines the different techniques that will help to secure transmitting data on cloud server. Ensures cloud Storage security. Overcomes all existing system, proposing the SecCloud and SecCloud+. SecCloud introduces an auditing entity with maintenance that helps clients to tag their file or data before uploading on server and to maintain the integrity of that datafile. SecCloud uses the proof-of-ownership protocol for secure data de-duplication also to prevent from data leakage on the internet. SecCloud+ is an advanced method for SecCloud that encrypts the clients data before uploading , and allows the secure integrity auditing and data de-duplication on that encrypted datafile. In proxy, the backup area the admin and users can directly take the datafiles.

REFERENCES

[1] M. D. Ryan, “Cloud computing security: The scientific challenge, and a survey of solutions,” Journal of Systems and Software, vol. 86, pp. 2263-2268, 2013.

[2] N. Kshetri, “Privacy and security issues in cloud computing: The role of institutions and institutional evolution,” Telecommunications Policy, vol. 37, pp. 372-386, 2013.

[3] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, et al., “Security and privacy for storage and computation in cloud computing,” Information Sciences, vol. 258, pp. 371-386, 2014.

[4] S. Khan, M. S. A. Khan, and C. S. Kumar, “Multi-criteria decision in the adoption of cloud computing services for SME’s based on BOCR Analysis,” Asian Journal of Management Research, vol. 5, pp. 621634, 2015.

[5] M. Sujithra, G. Padmavathi, and S. Narayanan, “Mobile device data security: a cryptographic approach by outsourcing mobile data to cloud,” Procedia Computer Science, vol. 47, pp. 480-485, 2015.

[6] A. Achuthshankar, A. Achuthshankar, K. Arjun, and N. Sreenarayanan, “Encryption of Reversible Data Hiding for Better Visibility and High Security,” Procedia Technology, vol. 25, pp. 216223, 2016.

[7] G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, “Scalable and efficient provable data possession,” in Proceedings of the 4th International Conference on Security and Privacy in Communication Netowrks, ser. SecureComm ’08. New York, NY, USA: ACM, 2008, pp. 9:1–9:10

[8] C. Erway, A. K¨upc¸ ¨u, C. Papamanthou, and R. Tamassia, “Dynamic provable data possession,” in Proceedings of the 16th ACM Conference on Computer and Communications Security, ser. CCS ’09. New York, NY, USA: ACM, 2009, pp. 213–222.

[9] F. Seb´e, J. Domingo-Ferrer, A. Martinez-Balleste, Y. Deswarte, and J.-J. Quisquater, “Efficient remote data possession checking in critical information infrastructures,” IEEE Trans. on Knowl. and Data Eng.,vol. 20, no. 8, pp. 1034–1038, 2008.

[10] E. Stefanov, M. van Dijk, A. Juels, and A. Oprea, “Iris: A scalable cloud file system with efficient integrity checks,” in Proc. 28th Annu. Comput. Secur. Appl. Conf., 2012, pp. 229–238.

[11] M. Azraoui, K. Elkhiyaoui, R. Molva, and M. €Onen, “Stealthguard: Proofs of retrievability with hidden watchdogs,” in Proc. Comput. Secur., 2014, pp. 239–256.

[12] J. Li, X. Tan, X. Chen, and D. Wong, “An efficient proof of retrievability with public auditing in cloud computing,” in Proc. 5th Int. Conf. Intell. Netw. Collaborative Syst., 2013, pp. 93–98.

[13] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, “Secure deduplication with efficient and reliable convergent key management,” IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 6, pp. 1615–1625, Jun. 2014.

[14] R. Di Pietro and A. Sorniotti, “Boosting efficiency and security in proof of ownership for deduplication,” in Proc. 7th ACM Symp. Inform., Comput. Commun. Secur., 2012, pp. 81–82.

[15] H. Wang, “Proxy provable data possession in public clouds,” IEEE Transactions on services Computing, vol. 6, no. 4, pp. 551–559, 2013.

[16] R. Di Pietro and A. Sorniotti, “Boosting efficiency and security in proof of ownership for deduplication,” in Proc. 7th ACM Symp. Inform., Comput. Commun. Secur., 2012, pp. 81–82.

[17] W. K. Ng, Y. Wen, and H. Zhu, “Private data deduplication protocols in cloud storage,” in Proc. 27th Annu. ACM Symp. Appl. Comput., 2012, pp. 441–446.

[18] J. Douceur, A. Adya, W. Bolosky, P. Simon, and M. Theimer, “Reclaiming space from duplicate files in a serverless distributed file system,” in Proc. 22nd Int. Conf. Distrib. Comput. Syst., 2002, pp. 617–624.

[19] M. Bellare, S. Keelveedhi, and T. Ristenpart, “Message-locked encryption and secure deduplication,” in Proc. Adv. Cryptol, 2013, pp. 296–312.

[20] M. Abadi, D. Boneh, I. Mironov, A. Raghunathan, and G. Segev, “Message-locked encryption for lock-dependent messages,” in Proc. Adv. Cryptol., 2013, pp. 374–391. [24] D. Boneh and M. Franklin, “Identity-based encryption from the Weil pairing,” in Proc. Adv. Cryptol., 2001, pp. 213–229.