International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 04 Issue-17
December 2017

Secure Auditing and Hybrid Deduplication Data in Cloud Computing systems

Srinivasa Rao Pathuri & D.Sobini
[1]M-TECH, Dept. of CSE,Swathi Institute of Technology & Sciences,Hyderabad, A.P., India
Mail Id: - srinivas.pathuri@yahoo.com
[2]HOD, Dept. of CSE,Swathi Institute of Technology & Sciences,Hyderabad, A.P., India
Mail Id: -sobini.b@gmail.com

Abstract

Cloud Storage are becoming more and more popular these days and hence they require the mechanism to down their utilization cost and provide more efficiency and security to client's data. Hybrid Deduplication is the technology which stores only a single copy of data and so it reduces the required storage space. In order to provide confidentiality to the client's data at the same time acquiring data deduplication, so we provide a technique called symmetric encryption. In this, data encrypted before outsourcing. To provide better data security this first talks the problem of authorized data Hybrid deduplication, also we have presented several security schemes for data deduplication construction in a Cloud Computing Technology Models. The proposed method shows that the users to achieve both data integrity and storage efficiency, results in non-trivial duplication of metadata (i.e., authentication tags), which contradicts the objectives of POW protocol implementation. Recent attempts to this problem introduce tremendous computational and communication costs and have also been proven not secure. Our Proposed Architecture allows Hybrid deduplication of both files and their corresponding authentication tags. Data integrity auditing and storage deduplication are achieved simultaneously. Analysis and Experimental results on different systems shows that our scheme is efficient and scalable.

Key words:-Hybrid Deduplication, Authentication, POW, Encryption, Data integrity,File Auditing.

INTRODUCTION

The Cloud storage is a model of networked enterprise storage .where data is stored in virtualized pools of storage which are generally hosted by third parties. Cloud Storage provides customers with benefits, ranging from cost saving and simplified

convenience, to mobility opportunities and scalable service. These great features attract more and more customers to utilize and storage their personal data to the cloud storage: according to the analysis report, the volume of data in cloud is expected to achieve 40 trillion gigabytes in 2020.Even though cloud storage system has widely adopted, it fails to accommodate some important emerging needs such as the abilities of auditing integrity of cloud files by cloud clients and detecting duplicated files by cloud servers. We illustrate both problems below.The First Problem is integrity auditing, the cloud server is able to relieve clients from the heavy burden of storage management and maintenance. The most difference of cloud storage from traditional in-house storage is that the data is transferred via internet and stored in an uncertain domain, not under control of the clients at all, which inevitably raises clients great concerns on the integrity of their data. These concerns originate from the fact that the cloud storage is susceptible to security threats from both outside and inside of the cloud, and the uncontrolled cloud servers may passively hide some data loss incidents from the clients to maintain their reputation. What is more serious is that for saving money and space, the cloud servers might actively and deliberately discard rarely accessed data files belonging to an ordinary client. Considering the large size of the outsourced data files and the client's constrained resource capabilities, the first problem is generalized as how can the client efficiently perform periodical integrity verifications even without the local copy of data files.The Second problem is secure deduplication. The rapid adoption of cloud services is accompanied by increasing volumes of data stored at remote cloud servers. Among these remote stored files, most of them are duplicated: according the recent survey by EMC, 75% of recent digital data is duplicated copies. This fact raises a technology namely deduplication, in which the cloud servers would like to deduplicate by keeping only a single copy for each file (or block) and make a link to the file (or block) for every client who owns or asks to store the same file (or block). Unfortunately, this action of hybrid deduplication would lead to a number of threats potentially affecting the storage system, for example, a server telling a client that it (i.e., the client) does not need to send the file reveals that some other client has the exact same file, which would be sensitive sometimes. These attacks originate from the reason that the proof that that client owns a

given file (or block of data) is solely based on a static, short value (in most cases the hash of the file). Thus, the second problem is generalized as how can the cloud server efficiently confirm that the client (with a certain degree assurance) owns the uploaded file (or block) before creating link to this file (or block) for them. The problem of integrity auditing and secure deduplication on cloud data. the second problem is generalized as how can the cloud servers efficiently confirm that the client (with a certain degree assurance) owns the uploaded file (or block) before creating a link to this file (or block) for him/her. This design fixes the issue of previous work that the computational load at user or auditor is too huge for tag generation. We specify that our proposed SecCloud system has achieved both integrity auditing and file deduplication. However, it cannot prevent the cloud servers from knowing the content of files having been stored. In other words, the functionalities of integrity auditing and secure deduplication are only imposed on plain files. In this section, we propose SecCloud+, which allows for integrity auditing and deduplication on encrypted files.

2. SCOPE OF WORK

2.1. Existing System

A number of deduplication systems have been proposed based on various hybrid deduplication strategies such as client side or server-side deduplications, file-level or block-level de-duplication. Ateniese et al. proposed a dynamic PDP schema but without insertion operation.Erway et al. improved Ateniese et al.'s work and supported insertion by introducing authenticated flip table.Wang et al. proposed proxy PDP in public clouds.Zhu et al. proposed the cooperative PDP in multi-cloud storage.•Wang et al. improved the POR model by manipulating the classic Merkle hash tree construction for block tag authentication.Xu and Chang proposed to improve the POR schema with polynomial commitment for reducing communication cost. Stefanov et al. proposed a POR protocol over authenticated file system subject to frequent changes. Azraoui et al. combined the privacy-preserving word search algorithm with the insertion in data segments of randomly generated short bit sequences, and developed a new POR protocol. Li et al. considered a new cloud storage architecture with two independent cloud servers for integrity auditing to reduce the computation load at client side.

2.2. Proposed System

In this scheme, aiming at achieving data integrity and hybrid dedupliation in cloud. We propose two secure systems namely SecCloud and SecCloud+.SecCloudintroduces an auditing entity with maintenance of a MapReduce cloud, which helps clients generate data tags before uploading as well as audit the integrity of data having been stored in cloud.Besides supporting integrity auditing and secure deduplication, SecCloud+ enables the guarantee of file confidentiality. We propose a method of directly auditing integrity on encrypted data. File ConfidentialityThe Design goal of file confidentiality requires preventing the cloud servers from accessing the content of files. Specially, we require the great that the goal of file confidentiality needs to be resistant to "dictionary attack". That event the adversaries pre-knowledge of the "dictionary" which includes all the possible files; they still cannot recover the target file.Specifically, data are split into fragments by using secure secret sharing schemes and stored at different servers.

3.IMPLEMENTATION

3.1 Cloud Clients

The data files to be stored and rely on the cloud for data maintenance and computation. They can be either individual consumers or commercial organizations.

3.2 Cloud Servers

It virtualize the resources according to the requirements of clients and expose them as storage pools. Typically, the cloud clients may buy or lease storage capacity from cloud servers, and store their individual data in these bought or rented spaces for future utilization.

3.3 Cloud Auditor/TPA

Auditor which helps clients upload and audit their outsourced data maintains a MapReduce cloud and acts like a certificate authority. This assumption presumes that the auditor is associated with a pair of public and private keys. Its public key is made available to the other entities in the system.

3.4 File Upload

This protocol aims at allowing clients to upload files via the auditor. Specifically, the file uploading protocol includes three phases.Phase2 (Cloud client -> auditor): Client upload files to the auditor, and receives a receipt from auditor.Phase3 (auditor -> Cloud KeyServer): auditor helps generate a set of tags for the uploading file, and send them along with this file to cloud server.

3.5 Integrity Auditing

It is an interactive protocol for integrity verification and allowed to ben initialized by

International Journal of Research
Available at https://edupediapublications.org/journals
e-ISSN: 2348-6848
p-ISSN: 2348-795X
Volume 04 Issue-17
December 2017

any entity except the cloud server. In this protocol, the cloud server plays the role of prover, while the auditor or client works as the verifier. This protocol includes two phases.

3.6 Proof of ownership

It is an interactive protocol initialized at the cloud server for verifying that the client exactly owns a claimed file. This protocol is typically triggered along with the file uploading protocol to prevent the leakage of side channel information. On the contrast to integrity auditing protocol, in PoW the cloud server works as verifier, while the client plays the role of prover. This protocol also includes two phases.

3.7 Data Integrity

Data Integrity is very important in database operations in general. Because Data Integrity ensured that data is of high quality, correct, consistent and accessible.

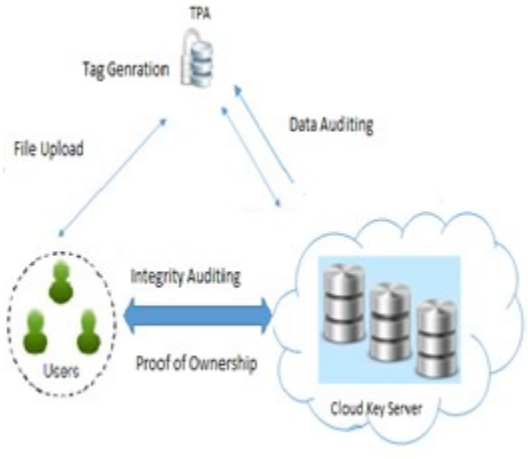

Fig 1 Architecture Diagram

4. EXPERIMENTAL RESULTS

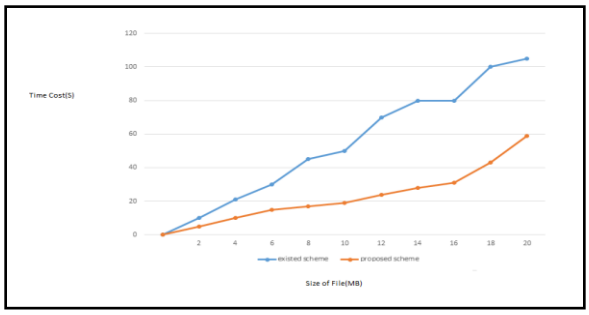

Fig 2: Tag generation

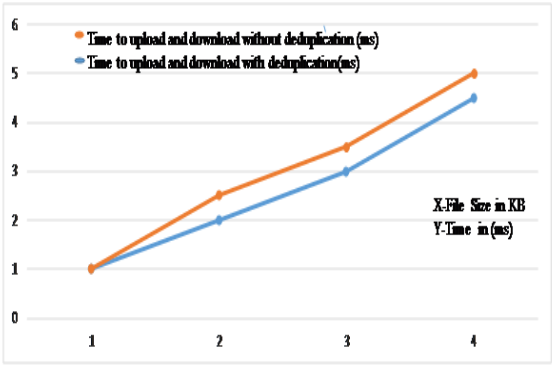

Fig 3: Comparison of duplicate and deduplicate files for uploading and downloading Time

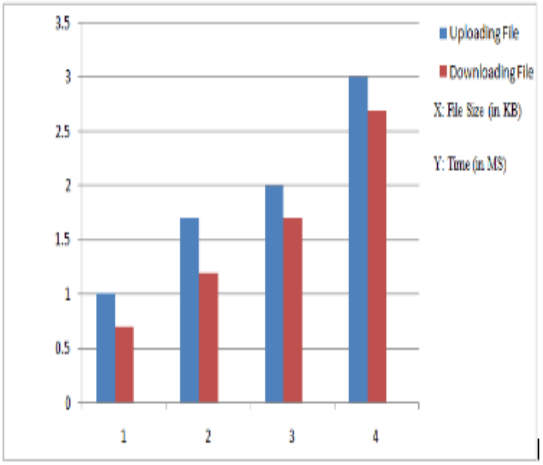

Fig 4:Time and Space Complexity with Deduplication

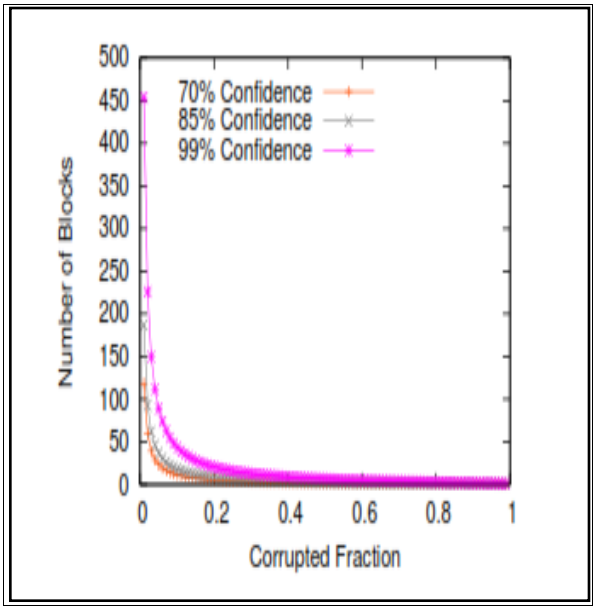

Fig 5:Corrupted Fraction

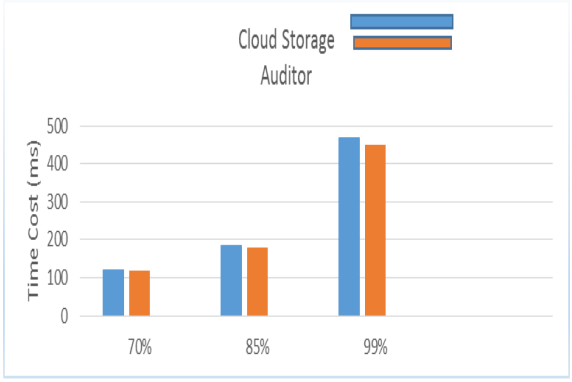

Fig 6: File Auditing

5. CONCLUSION

To achieve the data integrity as well as Hybrid data deduplication, Here we introduces two models that focused i.e., SecCloud Server and seccloud+ Server. The SecCloud Server is useful for client which generates the tags of source files. We introducing the Proof of Ownership Protocol for avoiding the leakage of side channel information. As compare to previous base paper works the computation time is decreased here. The SecCloud+ Server introduced the encryption mechanism for stored file for better security. In simple words it stores all files in encrypted format. In the network the data transferring of files between Cloud Servers different attacks will be possible. If further future scope new proposed security algorithms to implement at Hybrid Cloud Servers to better access of Cloud Storage Infrastructure Services.

ACKNOWLEDGEMENTS

This work was supported by National Natural Science Foundation of China (No.61100224 and No. 61472091), NSFCGuangdong(U1135002) and Natural Science Foundation of Guangdong Province (Grant No. S2013010013671).

6. REFERENCES

[1] Jingwei Li, Jin Li, DongqingXie and Zhang cai "Secure Auditing and Deduplication Data in Cloud" vol. 65, no. , pp. 2386-2396.

[2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," Communication of the ACM, vol. 53, no. 4, pp. 50–58, 2010.

[3] J. Yuan and S. Yu, "Secure and constant cost public cloud storage auditing with deduplication," in IEEE Conference on Communications and Network Security (CNS), 2013, pp. 145–153.

[4] S. Halevi, D. Harnik, B. Pinkas, and A. Shulman-Peleg, "Proofs of ownership in remote storage systems," in Proceedings of the 18th ACM Conference on Computer and Communications Security. ACM, 2011, pp. 491–500.

[5] S. Keelveedhi, M. Bellare, and T. Ristenpart, "Dupless: Serveraided encryption for deduplicated storage," in Proceedings of the 22Nd USENIX Conference on Security, ser. SEC'13. Washington, D.C.: USENIX Association, 2013, pp. 179–194. [Online].

[6] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in Proceedings of the 14th ACM Conference on Computer and Communications Security, ser. CCS '07. New York, NY, USA: ACM, 2007, pp. 598– 609.

[6] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, L. Kissner, Z. Peterson, and D. Song, "Remote data checking using provable data possession," ACM Trans. Inf. Syst. Secure., vol. 14, no. 1, pp. 12:1–12:34, 2011.

[7] J. Li, X. Tan, X. Chen, and D. Wong,"An efficient proof of retrievability with public auditing in cloud computing," in 5th International Conference on Intelligent Networking and Collaborative Systems (INCoS), 2013, pp. 93–98.

[8] J. Douceur, A. Adya, W. Bolosky, P. Simon, and M. Theimer, "Reclaiming
Space from duplicate files in a serverless distributed file system,"in 22nd International Conference on Distributed Computing Systems, 2002, pp. 617–624.

[9] M. Abadi, D. Boneh, I. Mironov, A. Raghunathan, and G. Segev,"Message-locked encryption for lock-dependent messages," in Advancesin Cryptology– CRYPTO 2013, ser. Lecture Notes in Computer Science. Canetti and J. Garay, Eds. Springer Berlin Heidelberg, 2013, vol.8042, pp. 374–391.

[10] 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.

[11] H. Wang, "Proxy provable data possession in public clouds," IEEE Transactions on Services Computing, vol. 6, no. 4, pp. 551–559, 2013.

[12] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," IEEE Transactions on Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231–2244, 2012.