Duplication Removal and Auditing Check Over Cloud

Pranita B. Wadavkar[1], Prashant M. Mane[2]

[1]Department of Computer Engineering, Zeal College of Engg. And Research, Savitribai Phule Pune University Pune, India

[2]Asst. Professor, Department of Computer Engineering, Zeal College of Engg. And Research, Savitribai Phule, Pune University Pune, India

Abstract: ***As the cloud computing development creates the most recent decade, outsourcing information to cloud service for storage turns into an attractive example, which benefits in saving efforts on substantial information maintain and manage, In any case, following the outsourced cloud storage is not completely reliable, it raises security worries on the most capable method to recognize information deduplication in cloud while accomplishing Integrity Auditing. In this work, the issue of integrity auditing and secure deduplication on cloud information. Specifically, going for accomplishing both information integrity and deduplication in cloud, in this propose two protected frameworks, to be specific SecCloud and SecCloud+ with additionally block level deduplication is achieve for saving the storage space of cloud. SecCloud presents an auditing substance with an maintain a MapReduce cloud, which assists clients with producing information tags before transferring and in addition audit the integrity of information having been store in cloud. Compared with past work, the calculation by client in SecCloud is significantly minimizes during the file transferring and reviewing stages. SecCloud+ is create to motivate by the way that clients constantly need to encode their information before transferring, and empowers honesty evaluating and secure deduplication on encoded information.***

Keywords: Integrity Auditing, Deduplication, Proof of Ownership, Convergent Encryption.

1. Introduction

Cloud storage is a framework of networked enterprise storage where information is stored in virtualized pools of the storage which are created by and large facilitated by third parties. Cloud storage gives clients with advantages, extending from cost saving and simplifies comfort, to versatility opportunities and adaptable service. These awesome components attract in more clients to use and store their own information to the Cloud storage: according to the analysis report, the volume of information in cloud is normal to accomplish 40 trillion gigabytes in 2020.

The main issue is integrity auditing. The cloud server is able customers from the larger burden of storage service and support. The most difference of cloud storage from traditional in-house storage is that the information is transferred by means of Internet and also stored in an uncertain domain, not under control of the customers by any means, which definitely raises customers extraordinary worries on the integrity of their information. These concerns start from the fact that the cloud storage is defenseless to security threats from both outside and inside of the cloud, and the uncontrolled cloud servers might passively hides some information loss incidents from the customers to keep up their reputation. In addition genuine is that for saving money and space, the cloud servers may even effectively and also, purposely dispose of rarely in a while got to data files having a place with a common customer. Considering the large size of the outsourced data files and the customers' constrained asset capacities, to begin with the issue is generalized as in what manner can the customer efficiently perform periodical integrity confirmations even without the local copy of data files. SecCloud presents an auditing entity with a support of a MapReduce cloud, which offers customers to create data tags before uploading and also review the honesty of information having been stored in cloud. For fulfillment of fine-grained, auditing function in the SecCloud are supported on both block level and sector level. What's more, SecCloud likewise enables secure deduplication. With a specific end goal to keep the storage of such side channel data, take after the custom of and design a proof of ownership protocol in between of customers and cloud servers, which permits customers to prove to cloud servers that they precisely own the objective data.

2. Related Work

2.1 Integrity Auditing

The provable data possession (PDP) was developed by Ateniese et al. [1][2] for encourage that the cloud servers holds the target files without retrieving or downloading the complete data. Ateniese et al. [3] develop a dynamic PDP schema but not support insertion operation; Erway et al. [4] increase Ateniese et al.'s work [5] and allows insertion by developing authenticated flie table; Same work has also used in [6]. Still, these plans suffer from the computation for tag generation at the client. To solve this problem, Wang et al. [7] developed proxy PDP in public clouds. Zhu et al. [8] developed the cooperative PDP in multi-cloud storage.

2.2 Secure Deduplication

secure deduplication target on the confidentiality of deduplicated data and considers to compose deduplication on encrypted data. Ng et al. [9] firstly developed the private data deduplication as a supliment of public data deduplication protocols of Halevi et al. [10]. Convergent encoding is a rising cryptographic primitive for assure data privacy in deduplication. Bellare et al. [11] developed this primitive as message-locked encryption, and analyze its application in able to secure outsourced storage. Abadi et al. [12] another justify Bellare et al's security assuming plaintext distributions that depend on the public parameters of the schemas. About the practical development of convergent encoding for securing deduplication, Keelveedhi et al. [13] designed the DupLESS system in which clients encrypt the file-based keys derived from a key server via an abstracted pseudorandom function protocol.

3. Proposed System

3.1 Description

System verify that system proposed SecCloud structure has achieved both integrity auditing and file deduplication. In any case, it can't keep the cloud servers from knowing the substance of files having been secured. At the end of the day, the functionalities of honesty examining and secure deduplication are simply constrained on plain files. Around there, system propose SecCloud+, which takes into account integrity auditing and deduplication on scrambled files. In this additionally take block level deduplication to save storage & save bandwidth of the cloud server. First file divide into block and then generate tag of each block, and then each block is duplicate or not is check by using block tag that is generated by block contents.

3.2 System Architecture

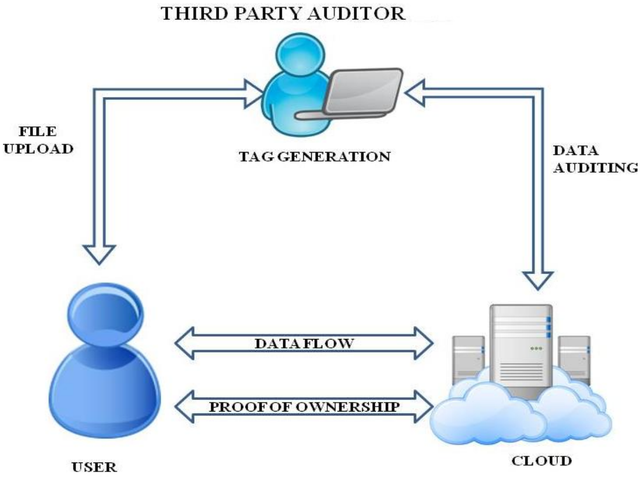

Figure1: System Architecture

Cloud client have large file to upload in cloud to store and maintain in cloud. Cloud server store user data in cloud. Auditor perform auditing for user to check whether user's files integrity is maintained on cloud or not for that here perform integrity auditing and proof of ownership for deduplicate files between clients and cloud server. TPA perform auditing tasks like challenge to server, get response from server and verify result.

3.3 Mathematical Model

Let S be the system object
It consist of following
S={U,F,TPA,CSP}
U= no of users
U={u1,u2,u3,…..un}
F= no of files
F={f1,f2,f3,…..fn}
B=no of blocks.
B{B1,B2,…,Bn}
TPA= Third Party Auditor
TPA={C,PF,V,POW}
C=challenge
PF =proof by CSP
V= verification by TPA
POW= proof of ownership
CSP= Cloud Service provider
CSP={PF,F}
PF=proof
F=files

3.4 Used Algorithms

3.4.1 KeyGen(F) :
The key generation algorithm gives a input as file content F and generate outputs as the convergent key ckF for F.

3.4.2 Encrypt(ckF;F) :
The encryption algorithm gives input as the convergent key ckF with file content F and generate the ciphertext ctF as output.

3.4.3 Decrypt(ckF; ctF) :
The decryption algorithm gives input, the convergent key ckF with ciphertext ctF and generate the plain file F as output.

3.4.4 TagGen(F) :
The tag generation algorithm gives input, a file content F and generate the tag tagF of F.

3.4.5 Advanced Encryption Standard :
The following AES steps of encryption for a 128-bit block are given below:
1. Derive the set of round keys from the cipher key.
2. Initialize the state array with the block data (plaintext).
3. Add the initial round key to the starting state array.
4. Perform nine rounds of state manipulation.
5. Perform the tenth and final round of state manipulation.
6. Copy the final state array out as the encrypted data (ciphertext).

3.4.6 MD5 :
MD5 algorithm used for generating file digests. It takes input message of arbitrary length and generates 128-bit long output hash. It first append padding bits and append length then initialize MD buffer and process message in 16-word blocks and finally get output.

3.5 Result Analysis

Table 1: Performance of secure Auditing

File Size	*File Upload*	*Integrity Auditing*	*Proof of Ownership*
10(KB)	0.05	0.48	0.01
50(KB)	1.75	0.92	0.02
100(KB)	2.5	1.38	0.035
200(KB)	4.8	1.94	0.5

4. Conclusion

In this Paper both data integrity and deduplication in the cloud are achieved, here propose SecCloud and SecCloud+. SecCloud represents an auditing entity with maintaining of a MapReduce cloud, that will be offers customers some of help with creating information tags before uploading and also audit the integrity of information having been stored in cloud. SecCloud enables secure deduplication through presenting a Proof of Ownership protocol and figuring the leakage of side channel information in data deduplication. Differentiated with the past work, the calculation by client in SecCloud is fundamentally reduced during the file uploading also, auditing stages. SecCloud+ is a advance development impacted by the way that clients always uploading, and takes into consideration integrity auditing also, secure deduplication directly on encoded data.

5. Acknowledgement

It gives me a great pleasure and immense satisfaction to present this special topic on “Duplication Removal And Auditing Check Over Cloud ”, The success of this topic has throughout depended upon an exact blend of hard work and unending co-operation and guidance, extended to me.

References

[1] Jingwei Li, Jin Li, Dongqing Xie and Zhang Cai,"Secure Auditing and Deduplicating Data in Cloud,"IEEE TRANSACTIONS on computers vol. pp no.99,2015.
[2] G. Ateniese, R. Burns, R. Curtmola, J. Herring,L. Kissner, Z. Peterson, and D. Song, “Provable data possession at untrusted stores,” in Proceedingsof the 14th ACM Conference on Computer and CommunicationsSecurity, ser. CCS ’07. New York, NY, USA: ACM, 2007, pp. 598–609.
[3] 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. Secur., vol. 14, no. 1, pp. 12:1–12:34, 2011.
[4] G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, “Scalable and efficient provable data possession,” in Proceedings of the 4thInternational Conference on Security and Privacy in Communication Netowrks, ser. SecureComm ’08. New York, NY, USA: ACM, 2008,pp. 9:1–9:10.
[5] 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.
[6] 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.
[7] H. Wang, “Proxy provable data possession in public clouds,” IEEE Transactions on Services Computing, vol. 6, no. 4, pp. 551–559, 2013.
[8] 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.
[9] W. K. Ng, Y. Wen, and H. Zhu, “Private data deduplication protocols in cloud storage,” in Proceedings of the 27th Annual ACM Symposium on Applied Computing, ser. SAC ’12. New York, NY, USA: ACM, 2012, pp. 441–446.
[10] 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.
[11] 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.
[12] M. Bellare, S. Keelveedhi, and T. Ristenpart, “Message-locked encryption and secure deduplication,” in Advances in Cryptology – EUROCRYPT 2013, ser. Lecture Notes in Computer Science, T. Johansson and P. Nguyen, Eds. Springer Berlin Heidelberg, 2013, vol. 7881, pp. 296–312.
[13] M. Abadi, D. Boneh, I. Mironov, A. Raghunathan, and G. Segev, “Message-locked encryption for lock-dependent messages,” in Advances in Cryptology – CRYPTO 2013, ser. Lecture Notes in Computer Science, R. Canetti and J. Garay, Eds. Springer Berlin Heidelberg, 2013, vol. 8042, pp. 374–391.
[14] 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.

Author Profile

Pranita Bhaskarrao Wadavkar, Pursuing M.E in Department of Computer engineering at Zeal Education Society’s Zeal College of Engineering & Research Savitribai Phule, Pune University Pune, India.

Prof. Prashant M.Mane, Asst. Professor,Department Of Computer Engineering at Zeal Education Society’s Zeal College Of Engineering & Research Savitribai Phule, Pune University Pune, India.