ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

Security of Data Stored in Cloud by Regenerating Code Method

[1]Akshata Sawant, [2]Aishwarya Unkule, [3]Komal Jangam, [4]Hasina Shaikh, [5]Prof. Uma bhokare
[1]-[4] B.E. student, [5] Professor,
[1] – [5] Department of Computer Science & Engineering,
Yashoda Techanical campus Shivaji Universit, Satara, India

Abstract: **- Nowadays many organizations, enterprises, and companies store their large amount of data into the cloud to reduce the efforts and cost needs for storage management. Also, cloud servers allow data users to retrieve their data when he needs it. But providing confidentiality, availability, integrity to this outsourced data are major security challenges in cloud computing. There is a chance of data loss due to unauthorized access or hacking. Existing data recovery methods for regenerating coded data requires data owner to always stay online and handle auditing as well as recovery of data which is impossible. This system proposes public auditing scheme in which every block of the file have its own hash code so that auditor can check the integrity of data stored on a cloud. Privacy preserved because data owner uploads files in an encrypted format which done using AES algorithm. If data corrupted or lost, then proxy server help to regenerate back a lost data by using regenerating code which has lower repair bandwidth. when a file distributed across servers regenerating code have the capability of repairing failed node by connecting to another node. Auditor ensures data is in safe mode and proxy solve regeneration problem so data owner free from online burden. The analysis shows that proposed system is highly efficient, secure with very low communication overhead.**

***Index Terms*—audit, cloud computing, proxy, regenerating code.**

I. INTRODUCTION

Cloud computing is a kind of internet based computing where shared resources, data, information are provided to computers and other devices on demand[1]. Cloud computing include on demand delivery of IT resources like memory, server, virtual machines via the internet with pay-as-you-go pricing. Cloud computing applications easier because they do not need to be installed on each user's computer and can be accessed from any different places. cloud-computing providers offer their services according to different models. There are three service models Software as a service (SAAS), Platform as a service(PAAS), Infrastructure as a service (IASS). It includes services like Email, games, database, development tools and other resources. There are four deployment models of cloud computing Public cloud, Private cloud, Hybrid cloud and Community cloud. In this way, Cloud computing, has now become a highly-demanded service or utility due to the advantages of high computing power, cheap cost of services, high performance, scalability, accessibility as well as availability[7], so individuals as well as industries use cloud storage to keep their data in cloud.

However, this cloud storage service also suffers from various security challenges. The Users are giving their data to the cloud which is not the trusted one and thus data can be corrupted, modified, stolen or even deleted. It is noted that data owners lose ultimate control over the fate of their outsourced data; thus, the correctness, availability and integrity of the data are being put at risk. on the other hand, the cloud service providers may act dishonestly, attempting to hide data loss or corruption and claiming that the files are still correctly stored in the cloud for reputation or monetary reasons. So, this Outsourcing the data on cloud introduced new security related problem like data loss and corruption so there is need to check data integrity as well as necessary to ensure correctness and availability of outsourced data.

To fully ensure the data integrity and minimize online burden, we propose a public auditing scheme for Regenerating-code-based cloud storage, in which the integrity checking done by a third-party auditor on behalf of the data owner. TPA perform periodically verification of their outsourced data to check the integrity of data[2]. Besides, to protect data privacy against the auditor data is uploaded in encrypted format so auditor cannot understand actual content. This method is lightweight and does not introduce any computational overhead to the cloud servers or TPA. Proxy is used to recover corrupted data. So, overhead of using cloud storage will be reduced as much as possible such that user does not need to stay online and to perform complex operation to their outsourced data.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

II. RELATED WORK

The problem of remote data checking for integrity was first introduced in Ateniese [3] . then in [21][8]. In PDP model[3] When a client stores data at untrusted server Provable data possession model allows the client to verify server possess original data without actually retrieving data from a server. also without accessing the whole file. In this scheme, the server generates probabilistic proofs of possession by sampling random set of blocks. The Client maintains metadata to verify this proof. For achieving public auditing, it uses RSA based homomorphic tags. It supports large data set & secure system for remote data checking. The Advantage of this scheme is data format independence & reduces I\O cost. But there is an overhead on the client to generate metadata of a file and It does not support dynamic auditing. In their subsequent work [10], they proposed a dynamic version of the prior PDP scheme based on MAC, which allows very basic block operations with limited functionality but block insertions. Simultaneously, Erway [21] gave a formal framework for dynamic PDP and provided the first fully dynamic solution to support provable updates to stored data using rank-based authenticated skit lists and RSA trees To release the data owner from online burden for verification, [3] considered the public auditability in the PDP model for the first time. However, their variant protocol exposes the linear combination of samples and thus gives no data privacy guarantee. Then [14], [15] developed a random blind technique to address this problem in their BLS signature based public auditing scheme. Similarly, Solomon et al. [10] presented a public PDP scheme, where the data privacy is provided through combining the cryptography method with the bilinearity property of bilinear pairing. Zhu et al. [11] proposed a formal framework for interactive provable data possession(IPDP) and **a** zero-knowledge IPDP solution for private clouds. Considering that the PDP model does not guarantee the retrievability of outsourced data, Juels [4] described a POR model, where spot-checking and error correcting codes are used to ensure both ”possession” and ”retrievability” of data files on remote archive service systems. A representative work upon the POR model is the CPOR presented by [11] with full proofs of security in the security model defined in [5]. They utilize the publicly verifiable homomorphic linear authenticator built from BLS Signatures to achieve public auditing. Furthermore ,In [9] proposed an efficient construction of cooperative provable data possession(CPDP) which can be used in multi-clouds, and [10] extend their primitive auditing protocol to support batch auditing for both multiple owners and multiple clouds.

III. SYSTEM MODEL

This System Model consist of four entities :
1.Data Owner- user who have a huge amount of data to store on cloud and access it.
2.Cloud- cloud service provider provide storage space, resources to store user’s data and maintain it.
3.TPA- perform auditing on coded data in the cloud without accessing the whole file.
4.Proxy-handle recovery by regeneration of data blocks

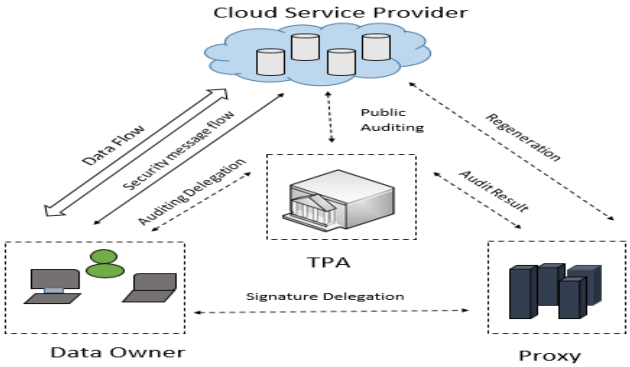

Figure 1: System Architecture

In this security of data stored on cloud by Regenerating code method While uploading data on cloud server data owner will split the file into several blocks and perform encryption of that blocks by using AES algorithm. Batch auditing and hash code is used to check integrity. Data privacy is preserved because neither the TPA nor the cloud server can see the actual contents because of encrypted format of uploaded files. This system consists of the proxy who find out the faulty server and help to regenerate the corrupted blocks.

IV. IMPLEMENTATION

Regenerating Codes:

Regenerating codes are first introduced by A. G. Dimakis et al. [18] for distributed storage to reduce the repair bandwidth. Viewing cloud storage to be a collection of n storage servers, data file F is encoded and stored redundantly across these servers. Then F can be retrieved by connecting to any k-out-of-n servers, which is termed the MDS2-property. When data corruption at a server is detected, the client will contact ℓ healthy servers and

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

download β′ bits from each server, thus regenerating the corrupted blocks without recovering the entire original file[2].

***Auditing Scheme*:**

Our auditing scheme consists of three procedures: Setup, Audit and Repair.

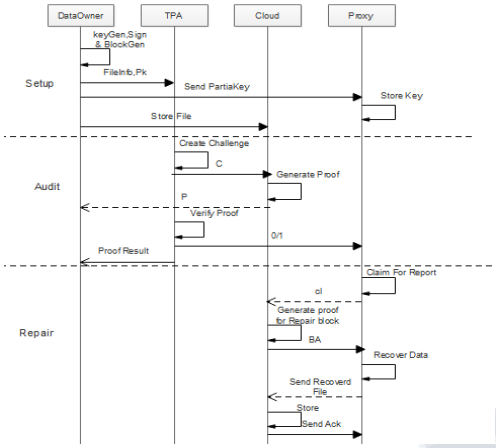

Figure 2.:Sequence chart of Scheme

1.Setup:data owner start system

KeyGen(1κ) → (pk,sk): This algorithm is run by the data owner to initialize its public and secret parameters by taking a security parameter κ as input.

Degelation(sk) → (x): In this data owner interact with proxy. The data owner send partial secret key x to the proxy.

SigAndBlockGen(sk,F) → (Φ,Ψ,t): This algorithm is run by the data owner and takes the secret parameter sk and the original file F as input, and then outputs a coded block set Ψ, an authenticator set Φ and a file tag t.

2.Audit: checking integrity

Challenge(Finfo) → (C): TPA with the information of the file Finfo as input and a challenge C as output.

ProofGen(C,Φ,Ψ) → (P): This algorithm is run by each cloud server with input challenge C, coded block set Ψ and authenticator set Φ, then it outputs a proof P.

Verify(P,pk,C) → (0,1): This algorithm is run by TPA immediately after a proof is received. Taking the proof P, public parameter pk and the corresponding challenge C as input, it outputs 1 if the verification passed and 0 otherwise.

***3.Repair*:** Regenerate Data

ClaimForRep(Finfo) → (Cr): It is similar with the above Challenge() algorithm in the Audit phase, but outputs a claim for repair Cr.as shown in figure2. GenForRep(Cr,Φ,Ψ) → (BA):The cloud servers run this algorithm upon receiving the Cr and finally output the block and authenticators set BA with another two inputs Φ,Ψ. BlockAndSigReGen(Cr,BA) → (Φ′,Ψ′,⊥): The proxy implements this algorithm with the claim Cr and responses BA from each server as input, and outputs a new coded block set Ψ′ and authenticator set Φ′ if successful, outputting ⊥ if otherwise.

Sequence of activities:

1. Data owner encrypt the data and store in the cloud.
2. At the same time data owner generate public key and private key.
4. Data owner send the partial secret key to the Third Party Auditor and Proxy.
5. Third Party Auditor contains the hash code of the file of data owner as well as file stored on the cloud.
6. TPA check the hash code for the original file and hash code of the cloud file. If Third Party Auditor found change in the hash code it immediately send acknowledgement to the proxy.
7. Then proxy agent replaces the changed file in the cloud.
8.If File is safe then data owner download it from cloud.

V. PERFORMANCE ANALYSIS

We calculate the performance of system during the setup, audit and repair phase. In setup phase file first encoded and then audit of every encoded block containing segment. Audit phase analyze the computational overhead and release those on cloud side. In repair phase regenerate the modified block and authenticator is represented to proxy for recovery.

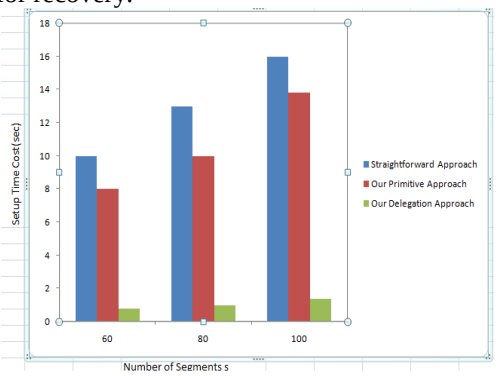

Figure 3.: Time for system setup with different

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

VI. SEGMENT NUMBERS

The above graph shows time required for system setup with different segment numbers. Here consider fixing parameters (n, k, l, α, β). n is the number of storage servers, k is the selective usable servers is the healthy servers, α is the storage cost and β is number of bits. In this graph consider three approaches such as straightforward approach, primitive approach and delegation approach. Straightforward approach requires nαs(m+3) operations reduced by primitive approach. It decrease the time of computation operation to nαs(m+1)+2m. But in our system, uses delegation approach reduces 1/18 of computational overhead of that with primitive approach which make the system more flexible.

Encryption and Decryption time:

The figure 4 graph shows encryption and decryption with respect to different file size. The behavior of the graph

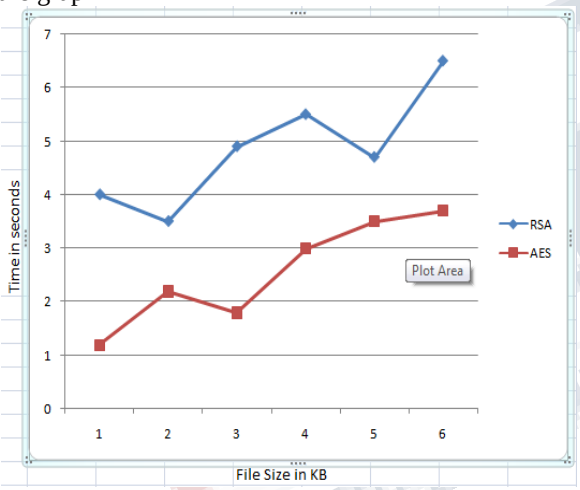

Figure5.: Server time Comparison

indicates if file size increased then time required for encryption and decryption also increased. In AES server time is required for encryption and decryption of file much less as compared to RSA algorithm. In this graph consider 120KB file the time needed by RSA system 4 to 7 second and AES is 1 to 4 second. In our system, we use AES to reduce server time required for encryption and decryption of file.

VII. CONCLUSION

In this way, we propose an auditing scheme to check the integrity of outsourced data as well as to recover corrupted data. This system provides a semi-trusted proxy to recover a data against corruption. The user can recover the failed data using the proxy server. We use TPA for auditing to check the integrity of data stored in cloud storage. TPA checks the data integrity on the behalf of data owner. Due to uploading files in the encrypted format, TPA cannot learn the data contents hence data is safe from TPA also. Performance analysis shows that our scheme is provable secure, more efficient and can be possible to integrate into a regenerating-code-based cloud storage system.

REFERENCES

[1] A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, and I. Stoica, "Above the clouds: A Berkeley view of cloud computing," Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley, Rep. UCB/EECS, vol. 28, p. 13, 2009.

[2] Jian Liu, Kun Huang, Hong Rong, Huimei Wang, and Ming Xian, "Privacy-Preserving Public Auditing for Regenerating -Code-Based Cloud Storage," in IEEE Trans. on information forensics and security , vol.. 10,no. 7, July 2015.

[3] 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.

[4] A. Juels and B. S. Kaliski Jr, "Pors: Proofs of retrievability for large files," in Proceedings of the 14th ACM conference on Computer and communications security. ACM, 2007, pp. 584–597.

[5] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "Mr-pdp: Multiple replica provable data possession," in Distributed Computing Systems, 2008. ICDCS'08. The 28th International Conference on. IEEE, 2008, pp. 411–420.

[6] K. D. Bowers, A. Juels, and A. Oprea, "Hail: a high-availability and integrity layer for cloud storage," in Proceedings of the 16th ACM conference on Computer and communications security. ACM, 2009, pp. 187–198.

[7] J. He, Y. Zhang, G. Huang, Y. Shi, and J. Cao, "Distributed data possession checking for securing

multiple replicas in geographically dispersed clouds," Journal of Computer and System Sciences, vol. 78, no. 5, pp. 1345–1358, 2012.

[8] Chen, R. Curtmola, G. Ateniese, and R. Burns, "Remote data checking for network coding-based distributed storage systems," in Proceedings of the 2010 ACM workshop on Cloud computing security workshop. ACM, 2010, pp. 31–42.

[9] H. Chen and P. Lee, "Enabling data integrity protection in regenerating coding-based cloud storage: Theory and implementation," Parallel and Distributed Systems, IEEE Transactions on, vol. 25, no. 2, pp. 407–416, Feb 2014.

[10] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," Parallel and Distributed Systems, IEEE Transactions on, vol. 24, no. 9, pp. 1717–1726, 2013.

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

[12] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," Proceedings of the IEEE, vol. 99, no. 3, pp. 476– 489, 2011. [12] H. Shacham and B. Waters, "Compact proofs of retrievability," in Advances in Cryptology-ASIACRYPT 2008. Springer, 2008, pp. 90– 107.

[13] Y. Hu, H. C. Chen, P. P. Lee, and Y. Tang, "Nccloud: Applying network coding for the storage repair in a cloud-of-clouds," in USENIX FAST, 2012

[14] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in INFOCOM, 2010IEEE, 2010, pp. 1–9.

[15] C.Wang,S.S.Chow,Q.Wang,K.Ren,andW.Lou"Privacy -preserving public auditing for secure cloud storage," Computers, IEEE Transactions on, vol. 62, no. 2, pp. 362–375, 2013

[16] Yogesh Shinde, Omprakash Tembhurne "A review of protect the integrity of outsourced data using third party auditing for secure cloud storage," Computers, International Journal of Science and Research(IJSR).

[17] Mr.Satish Shelar1, Prof.S.Y.Raut2" Review On Regenerating Code Based Secure Cloud Storage Using Public Auditing " International Research Journal of Engineering and Technology (IRJET) Volume: 02 Issue: 09 | Dec-2015.

[18] Yogesh Shinde , Alka Vishwa" Privacy Preserving using Data Partitioning Technique for Secure Cloud Storage" International Journal of Computer Applications (0975 – 8887) Volume 116 – No. 16, April 2015.

[19] Madhumati B.Shinde1, S. B. Sonkamble " Survey on Secure Public Auditing and Privacy Preserving in Cloud" (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 6 (1) , 2015, 10-13

[20] Jyoti R Bolannavar1" Privacy-Preserving Public Auditing using TPA for Secure Cloud Storage" International Journal of Scientific Engineering and Research (IJSER) Volume 2 Issue 6, June 2014

[21] 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. ACM, 2009, pp. 213–222.