ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 6, June 2015

Efficient Cloud Computing with Secure Data Storage using AES

Mr. Santosh P. Jadhav[1], Prof. B. R. Nandwalkar[2]

Department of Computer Engineering, Late G.N.Sapkal Collage of Engineering, University of Pune, India[1,2]

Abstract: Most of the security schemes in cloud environment had not addressed the privacy preserving between third party auditor and the data in the cloud. Encryption techniques used previously are RSA based, which have different loopholes which can be overcome by using the most prominent encryption techniques which uses advanced encryption standards (AES) encryption algorithm. AES is most frequently used encryption algorithm today this algorithm is based on several substitutions, permutations and linear transformations, each executed on data blocks of 16 byte. As of today, no practicable attack against AES exists. Therefore, AES remains the preferred encryption standard for governments, banks and high security systems around the world. For efficient auditing the formation of batch of task and are executed in batch wise fashion also increase efficiency of TPA, the batch auditing protocol. In cloud the in the cloud is not only accessed by the user but also update data frequently. hence, providing data dynamics in cloud computing is also of prominent importance

Keywords: RSA, AES, TPA.

I. INTRODUCTION

This Most of the security schemes in cloud environment had not addressed the privacy preserving between third party auditor and the data in the cloud. Encryption techniques used previously are RSA based, which have different loopholes which can be overcome by using the most prominent encryption techniques which uses advanced encryption standards(AES) encryption algorithm. AES is most frequently used encryption algorithm today this algorithm is based on several substitutions, permutations and linear transformations, each executed on data blocks of 16 byte. As of today, no practicable attack against AES exists. Therefore, AES remains the preferred encryption standard for governments, banks and high security systems around the world. For efficient auditing the formation of batch of task and are executed in batch wise fashion also increase efficiency of TPA, the batch auditing protocol. In cloud the in the cloud is not only accessed by the user but also update data frequently. Hence, providing data dynamics in cloud computing is also of prominent importance.

II. LITERATURE SURVEY

An Ateniese et al to consider public auditability in their defined provable data possession (PDP) model for ensuring possession of data files on untrusted storages. They implemented the scheme which utilizes the RSA-based homomorphic authenticators for auditing outsourced data and suggests randomly sampling of a few blocks of file. Though, the public auditability in their scheme demands the linear combination of sampled blocks exposed to external auditor. The protocol which they proposed when used directly, is not privacy preserving, and therefore, leaks user data information to the auditor.

2) Secondly, Juels et al.[4] described in proof of retrievability (PoR) model, where spot checking and error correcting codes are used to ensure both possession and irretrievability of data files on remote archive service systems. However, the number of audit challenges a user can perform is a permanent priori, and public auditability is not supported in their main scheme. Although they describe a straight forward Merkle-tree construction for public PoRs, this approach only works with encrypted data. Shacham et al.[12] Design an improved PoR scheme built from BLS signatures with full proofs of security. Similar to the construction i they use publicly verifiable homomorphic authenticators that are built from provably secure BLS signatures [19]. Public retrievability is achieved based on the elegant BLS construction. yet again, their approach does not support privacy-preserving auditing for the same reason as Shah et al.propose allowing a TPA to keep online storage onest by first encrypting the data then sending a number of pre-computed symmetric-keyed hashes over the encrypted data to the auditor. The auditors verify both the integrity of the data file and the server's control of a previously committed decryption key. This scheme only works for encrypted files and it suffers from the auditor easy way to comply with the conference paper formatting requirements is to use this document as a template and simply type your text into it. State fullness and bounded usage, which may potentially bring in on-line burden to users when the keyed hashes are used up.

3) Ateniese etal. Propose a partially dynamic version of the prior PDP scheme that uses only symmetric key cryptography. However, the system imposes a priori bound on the number of audits and does not support public auditability. Consider a similar support for partial dynamic data storage in distributed scenario. The proposed challenge-response protocol can both determine the data correctness and locate possible errors.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 6, June 2015

In a subsequent work, Wang et al. propose to combine BLS based homomorphic authenticator with MHT to support both public auditability and fully data dynamics.

4) Simultaneously, Erway et al.[16] developed a skip lists based scheme to enable provable data control with fully dynamics support. However, all their protocol requires the linear combination of sampled blocks just as, and thus does not support privacy- preserving auditing on users outsourced data. While all above schemes provide methods for efficient auditing and provable assurance on the correctness of remotely stored data, none of them meet all the requirements for privacy-preserving public auditing in Cloud Computing, as supported in our result. More importantly, none of these schemes consider batch auditing, which will greatly reduce the computation cost in the TPA when coping with large number of audit delegations.

5) Portions of the work presented in the paper[1] they have previously appeared as an extended abstract they have revised the paper[2] a lot and improved many technical details as compared The primary improvements are as follows: First, they provide a new privacy-preserving public auditing protocol with enhanced security strength in we also include an additional (but slightly less efficient) protocol design for provably secure zero- knowledge leakage public auditing scheme Second, based on the enhanced main auditing scheme, they provide a new provably secure batch auditing protocol. All the experiments in their performance evaluation for the newly designed protocol are completely redone. they extended main scheme to support data dynamics and provide discussions on how to generalize their privacy-preserving public auditing scheme in which are lacking in [2]. Finally, they provide formal analysis of privacy-preserving guarantee and storage correctness, only heuristic arguments are sketched in [2]. To securely introduce an ef-fective third party auditor (TPA), the following two fundamental requirements have to be met:

1) TPA should be able to efficiently audit the cloud data storage without demanding the local copy of data, and introduce no additional on-line burden to the cloud user;

2) The third party auditing process should bring in no new vulnerabilities towards user data privacy.

A. Frequently used algorithms

The Four basic algorithms are used frequently to set up the system environment.

1) Key generation
2) Sign Generation
3) Gen proof
4) Verify proof, which user uses key generation algorithm to set up the sheme. Verification metadata is generated by the sign generation algorithm, where signature or identity of user is generated. Genproof algorithm is run on the cloud server to check the data storage correctness in the cloud, and for auditing the proof TPA uses to audit the proof. Algorithms for preserving privacy between the user and the cloud. Homomorphic Linear Authenticator (HLA) with random masking technique is used. This teniques guarantee that during auditing process TPA will not demand for the local copy of data and will not be able to learn any knowledge about the data. Algebric properties of the authenticator are taken in such a manner that they are helpful for batch processing and auditing process in further extentionhe entire document should be in Times New Roman or Times font. Type 3 fonts must not be used. Other font types may be used if needed for special purposes.

III. IMPLEMENTATION DETAILS

We consider a cloud data storage service involving three different entities, as illustrated in the cloud user, who has large amount of data files to be stored in the cloud; the cloud server, which is managed by the cloud service provider to provide data storage service and has significant storage To fully ensure the data integrity and save the cloud users computation resources as well as online burden, it is of critical importance to enable public Auditing service for cloud data storage, so that users may resort to an independent third party auditor (TPA) to audit the outsourced data when needed. The TPA, who has expertise and capabilities that users do not, can periodically check the integrity of all the data stored in the cloud on behalf of the users, which provides a much more easier and affordable way for the users to ensure their storage correctness in the cloud. Moreover, in addition to help users to evaluate the risk of their subscribed cloud data services, the audit result from TPA would also be beneficial for the cloud service providers to improve their cloud based service platform, and even serve for independent arbitration purposes. In a word, enabling public auditing services will play an important role for this nascent cloud economy to become fully established; where users will need ways to assess risk and gain trust in the cloud.

Proposed Architecture

To enable privacy-preserving public auditing for cloud data storage under the above mentioned model, our protocol design should accomplish the following security and performance guarantee:

1) Public auditability: allow TPA to verify the correctness of the cloud data on demand without retrieving a copy of the whole data or introducing additional on-line burden to the cloud users.

2) Storage correctness: guarantee that there exists no cheating cloud server that can pass the audit from TPA without indeed storing users data intact.

3) Privacy-preserving: guarantee that there exists no way for TPA to derive users data content from the information collected during the auditing process.

4) Batch auditing: enable TPA with secure and efficient auditing capability to cope with multiple auditing delegations from probably large number of different users simultaneously.

5) Lightweight: allow TPA to perform auditing with minimum communication and computation overhead. We are introducing an attacking module which will keep continuously track on the data alteration in the cloud if any, and will inform the user about the altered data.

DOI 10.17148/IJARCCE.2015.4682

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 6, June 2015

Attacking module will be in the form of small code to modify the database directly so that entry is sabotaged. This code will reside on cloud server. Also the timer is going to be implemented where task may be schedule for one time execution, or for repeated execution at regular intervals and also we adapt some efficient servers for better performance and increase the speed of execution, such as glassfish server.

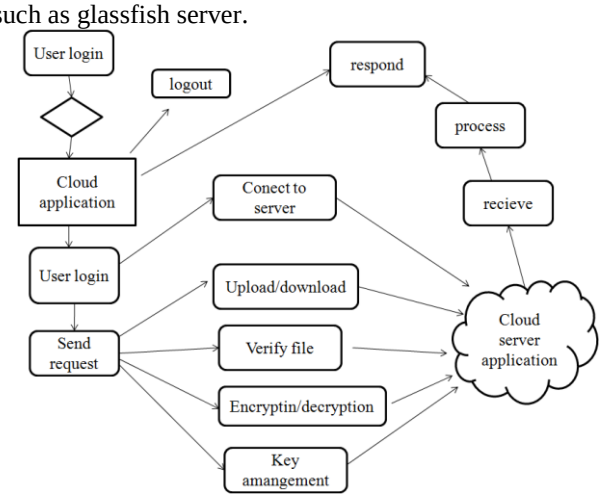

Fig1 architecture diagram

B. Privacy Preserving module

Homomorphic authenticators are unforgeable verification etadata generated from individual data blocks, which can e securely aggregated in such way to guarantee an auditor that a linear combination of data blocks is appropriately computed by verifying only the aggregated authenticator. Hence, to achieve privacy-preserving public auditing, we propose to uniquely integrate the homomorphic authenticator with random mask technique. In our protocol, the linear combination of sampled blocks in the server response is masked with randomness generated by a pseudo random function (PRF) [9].

C. Batch auditing module

Through the organization of privacy-preserving public auditing in Cloud Computing, TPA may concur rently handle multiple auditing delegations upon differ ent user requests. The individual auditing of these tasks for TPA can be and very difficult and inefficient. Batch auditing not only allows TPA to perform the multiple auditing tasks at the same time, but also greatly reduces the computation cost on the TPA side This is because of aggregating K verification equations into helps to reduce the number of quite expensive paring operation from 2k, as required in individual auditing ,to K+1, by which saves a considerable amount of auditing time[9].

Data dynamic support is achieved by replace information index in computation of block authenticator and by using one of the best data structure i.e. MHT (merkle hash tree). Supporting data dynamics for privacy-preserving public risk auditing is also of supreme importance. Now we show how our main scheme can be adapted to build upon the obtainable work to support data dynamics, including block level operations of modification, deletion and insertion. We can accept this technique in our design to achieve privacy-preserving public risk auditing with support of data dynamics. [9]

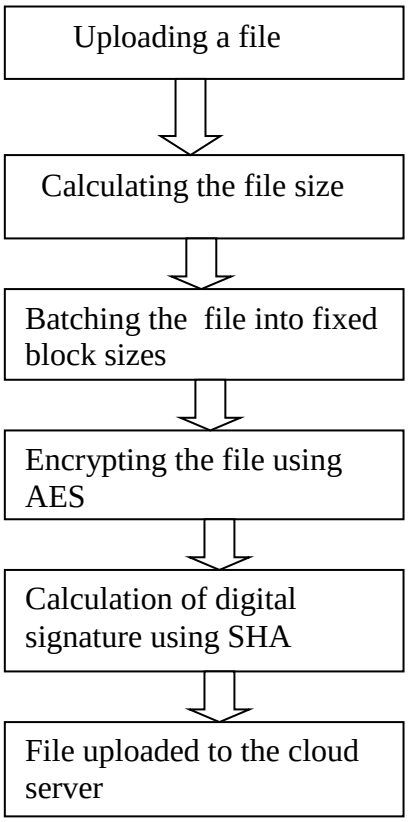

Fig2 steps in uploading the file

D. Verify module

This module verify that whether file is intrupted or modified and notify user accordingly by providing alert messages., also log records of the file alteration are also recorded for user point of view. Verify module keeps tracking the cloud data transaction in given time duration. we had also tried to show the security by shoe the file alteration by attacker module. Attacker module will alter the content of the data file and after executing the verify module this alteration is identified and tracked, and Hence avoid the file from downloading; we strongly can say that the system is safe.

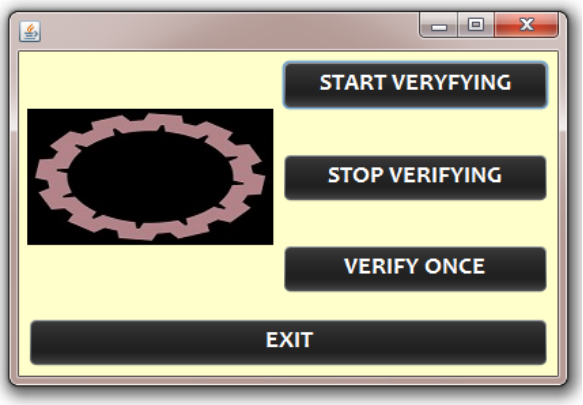

Fig3 steps in uploading the file

E. Use of AES

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 6, June 2015

We had implemented all the algorithms using AES encryption techniques which were previously implemented as RSA based encryption tecniquies the benefits of using AES are as fallow Some factors that are analysed by considering packet size while using AES, by which we expect that our sys tem will give efficient result then past developed system. Hence, AES encryption and decryption speed is much more less and therefore said to be more efficient then RSA. And many more benefits of using AES are mentioned in[13].

S.no	Factors analyzed	AES	RSA
1	Key length	256	1024
2	Simulation speed	high	low
3	Power consumption	low	high
4	Hardwareand software implementation	Highly efficient	Not efficient
5	Security	Highly secure	Min attack

Fig4 Camparisions of AES and RSA[13]

F. Use of SHA

SHA stands for secure hashing algorithm it produces 20 bytes160 bit hash value.which gives message uthentication and preserves the integrity during transaction. Authentication Requirements:
1)Masquerade – Insertion of message from fraudulent source
2) Content Modification – Changing content of message
3)Sequence Modification – Insertion, deletion and reordering sequence.
4) Timing Modification – Replaying valid sessions.

IV. RESULT ANALYSIS

After implementation of AES rather than RSA along with secured hash function we got more efficient result. With the best encryption technique algorithm i.e AES. System should show the efficient performance in its execution, the privacy preserving should be achieved so, that TPA should not demand the copy of whole data and will not any knowledge from the data or putting more burden on the end user. Auditing of data with batch wise processing is now scheduled with the help of timer that we have introduced newly. The performance of the system is improved by using glassfish server which is easy to handle and and has higher processing capabitlites. Attacking module used should be able to find Found that compared to individual auditing, batch auditing indeed helps reducing the TPA computation cost by 20 the altered data in the cloud when the data is stored or updated dynamically. As there are less number of expensive operation required for batching such as modular exponentions and multi plications. After conducting batch auditing test with increased no of task from 1 to 2000, with intervals of 8. It was percent. We had also tried to support data dynamics along with privacy preserving.

Total file sizes(kb)	No of blocks of the file(4kb)	Total uploading time using AES(ms)	Total uploading time using RSA (ms)
6.49	2	35181	37586
160	41	18838	33802
628	158	101179	156283

Fig5 Result table

Some factors that are analysed by considering packet size while using AES, by which we expect that our system will give efficient result then past developed system. Hence, AES encryption and decryption speed is much more less and therefore said to be more efficient then RSA. And many more benefits of using AES are mentioned in[13] The graphical reprentation of the result are shown in the following grap in wich uploading time is represented on y-axis,the blue line (Series1)represent the values for aes while red (series2) shoes value for RSA ,while data size is represented on x-axis,it shows that graph of RSA goes high which indicates the required more time for uoploading the file then AES. Results are taken on the system which has the following configuration intel core i3 processor,1.66 GHz spped,32 bit operating system,2gb RAM ,500gb hard disk. Result may vary on different configuration.

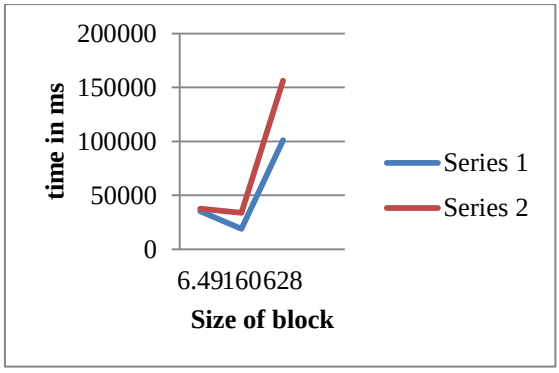

Fig6 graph block size vs time(ms)

V. CONCLUSION

In this paper, we propose a privacy-preserving public auditing system for data storage security incloud computing. Although the computational time is increased but the privacy is preserved. Where data is stored in the cloud by using the most prominent algorithm AES. We utilize the homomorphic linear authenticator and random masking to guarantee that the TPA would not learn any knowledge about the data content stored on the cloud server during the efficient auditing process, which not only eliminates the load of cloud user from the tedious and possibly expensive auditing task, but also reduces the users fear of their outsourced data leakage.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 6, June 2015

Considering TPA may concurrently handle multiple audit sessions from different users for their outsourced data files, we further extend our privacy-preserving public auditing protocol into a multi-user setting, where the TPA can perform multiple auditing tasks in a batch manner for better efficiency. We had overcome most of drawbacks of the existing system by securing data dynamics and performance improvement. General analysis shows that our schemes are provably secure and highly efficient. Our preliminary experiment conducted case further demonstrates the fast performance of our design on both the cloud and the auditor side. We leave the full-fledged implementation of the mechanism on commercial public cloud as an important future scope.

REFERENCES

[1] Cong Wang, Member, IEEE, Sherman S.M. Chow, Ian Wang, Member, IEEE, Kui Ren, Senior Mem-ber, IEEE, and Wenjing Lou, Senior Member, IEEE. "privacy preserving and public auditing in secure cloud storage" IEEE transaction on 2013.

[2] P. Oreizy, N. C. Wang, Q. Wang, K. Ren, and W.Lou. "Privacy Preserving Public Auditing for Storage Security in Cloud Computing" proc.IEEE INFOCOM 10, Mar 2010.

[3] Q.Wang, C.Wang, K. Ren, W. Lou, and J. Li. "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing" IEEE Trans.Parallel and Distributed Systems vol. 22, no. 5, pp. 847-859, May 2011.

[4] K.D.Bowers, A. Juels, and A.Oprea. "Proofs of Retrievability: Theory and Implementation" Proc. ACMWorkshop Cloud Computing Security (CCSW09).pp. 43-54, 2009

[5] P. Mell and T. Grance. "Draft NIST Working Definition of Cloud Computing"Availableat:hhttp://csrc.nist.gov/groups/SNS/cloudcomputing/index.html"2009.

[6] Cloud Security Alliance,"Top Threats to Cloud Computing" http://www.cloudsecurityalliance.org 2010.

[7] Cloud security alliance" security guidance for critical areas of focus in cloud computing" Available at: http://www.cloudsecurityalliance.org. Inc. Amazon Spot-Instances http://aws.amazon.com/ec2/spotinstances/ Dec.2009.

[8] S.H. Clearwater, R. C.Wang, Q.Wang, K. Ren, and W. Lou. "Towards Secure and Dependable Storage Services in Cloud Computing" IEEE Trans. Service Computing, vol. 5, no. 2, 220-232, Apr.-June

[9] Q.Wang, C.Wang, K. Ren, W. Lou, and J. Li. "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing" IEEE Trans Parallel and Distributed Systems, vol.22, no. 5, pp.847-859, May 2011.

[10] F. Sebe, J. Domingo-Ferrer , A. MartAśnez-Balleste , Y. Deswarte, and J.J. Quisquater. "Efficient Re-mote Data Possession Checking in Critical Information Infrastructures " IEEE Trans. Knowledge and Data Eng. vol. 20, no. 8, pp. 1034-1038, Aug. 2008.

[11] M. Stonebraker, R. Devine, M. Kornacker, W.Litwin, A. Pfeffer, A. Sah, and C. Staelin,Proc.Third M.A. Shah, R. Swaminathan, and M. Baker."Privacy-Preserving Audit and Extraction of Digital Contents " Cryptology ePrint Archive, Report 2008/186, 2008.

[12] H. Shacham and B. Waters. "Compact Proofs of Rerievability" Proc. Int Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (Asiacrypt) vol. 5350, pp. 90107. Dec. 2008.

[13] " International journel of science and research india online" ISSN:2319-7064

[14] D. Boneh, B. Lynn, and H. Shacham, "Short Signatures from the Weil Pairing" J. Cryptology, vol. 17,no. 4, pp. 297-319, 2004.

[15] M. A. Shah, M. Baker, J.C. Mogul, and R. Swaminathan, "Auditing to Keep Online Storage Services Honest" Proc. 11th USENIX Workshop Hot Topics in Operating Systems (HotOS 07), pp. 1-6, 2007

[16] C. Erway, A. Kupcu, C. Papamanthou, and R.Tamassia, "Dynamic Provable Data Possession"Proc. ACM Conf. Computer and Comm. Security (CCS 09), pp. 213-222, 2009.

[17] A. Juels and J. Burton, S. Kaliski, "PORs: Proofsof Retrievability for Large Files" Proc. ACM Conf.Computer and Comm. Security (CCS 07), pp. 584597, Oct. 2007.