ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 1, pp: (141-146), Month: January - March 2015, Available at: www.researchpublish.com

Privacy-Preserving On Cloud

Matale Sujata[1], Mogal Pallavi [2], Mogal Vruttika[3]

Department Of Computer Engineering, NDMVPS's KBTCOE, NASHIK, INDIA

Abstract: **Now a day's many IT organisation wish to go for cloud computing environment, as there are various advantages of preferring this environment. Many techniques and methods are used to build this cloud environment in order to provide secure access to cloud. Cloud user become free from different data management task, as this data management is done by the third party expert auditor (TPA). But still there are certain security issues of this third party auditor as during this audit TPA can get access to the data in the cloud. Security in cloud has become the most on demand issue to be addressed in cloud computing environment. Various method and techniques are used to provide security in cloud environment. Privacy preserving between cloud and TPA has to be addressed to avoid the data leakages. RSA based homomorphism authenticator was used to encrypt the data. Here, AES instead of RSA is proposed, to increase the efficiency. To speed up the auditing by TPA batch auditing scheme is introduced, which has the ability to audit the files batch wise. Supporting data dynamics is privacy preserving and public auditing has special importance. Care must be taken so that no new vulnerabilities should get introduced.**

Keywords: **AES, Batch auditing, public audit ability, privacy preserving, cloud computing, data storage.**

1. INTRODUCTION

Cloud Computing, which provides Internet based service and use of computer technology. This is cheaper and more strong processors, together with the software as a service (SaaS) computing architecture, are transforming data into data centres on huge scale. The increasing network and flexible network connections make it even possible that users can now use high quality services from data and provides remote on data centres. Storing data into the cloud offers great help to users since they don't have to care about the problems of hardware problems. While these internet-based online services do provide huge amounts of storage space and customizable computing resources, this computing platform shift, however, is avoids the responsibility of local machines for data maintenance at the same time.

As a result, users are at the interest of their cloud service providers for the availability and integrity of their data the one hand; although the cloud services are much more powerful and reliable than personal computing devices and broad range of both internal and external threats for data integrity still exist. Examples of outages and data loss incidents of noteworthy cloud storage services appear from time to time. On the other hand, since users may not keep a local copy of outsourced data, there exist various incentives for cloud service providers (CSP) [1] to behave unfaithfully towards the cloud users regarding the status of their outsourced data. Our work is among the first few ones in this field to consider distributed data storage security in Cloud Computing.

2. LITERATURE SURVEY

Juels et al. [4] proposed PoR model, i.e. the "Proof of Retrievability". In this model spot-checking and error correcting codes are used to ensure both possession as well as retrievability of data files on remote archive service systems. However, the public auditability is not supported in their main scheme and also the number of audit challenges a user can perform is fixed. Even though they describe a straightforward Merkle-tree construction for public PoRs, their approach works only for the encrypted data. The study on different variants of PoR with private auditability was given by Dodis et al. An improved PoR scheme built with full proofs of security in the security model was designed by Shacham et al. [5], [7]. Similar to the construction, use publicly verifiable homomorphic non-linear authenticators that are built from provably secure BLS signatures. A compact and public verifiable scheme is obtained based on the elegant BLS

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 1, pp: (141-146), Month: January - March 2015, Available at: www.researchpublish.com

construction. Again, their approach does not support privacy preserving auditing. The propose scheme allows the TPA [2] to keep online storage honestly by first encrypting the data then sending a number of pre computed symmetric-keyed hashes over the encrypted data to the auditor. The auditor then verifies the integrity of the data file as well as the server's possession of a previously committed decryption key. This scheme only works for encrypted files and it suffers from the auditor state fullness and bounded usage. This drawback may potentially bring an online burden to users when the keyed hashes are used up. Consider a similar support for partial dynamic data storage in a distributed scenario with additional feature of data error localization. In a subsequent work, Wang et al. [2] propose to combine BLS-based HLA with MHT to support both public auditability and full data dynamics. However, the linear combination of sampled blocks is required for the verification in these two protocols and thus does not support privacy preserving auditing. Methods for efficient auditing and provable assurance on the correctness of remotely stored data are being provided by all the above schemes whereas none of them fulfil all the requirements for privacy preserving public auditing in cloud computing. Along with it none of these schemes consider batch auditing, which can greatly reduce the computation cost on the TPA when coping with a large number of audit delegations.

3. PROBLEM DEFINITION

Consider a cloud data storage service involving three different entities, the cloud user (U), who has large amount of data files to be stored in the cloud; the cloud server (CS), which is managed by the cloud service provider (CSP) to provide data storage service and has significant storage space and computation resources, the third party auditor (TPA), who has expertise and capabilities that cloud users do not have and is trusted to assess the cloud storage service reliability on behalf of the user upon request. For the storage and maintenance of the data on cloud, the users rely on the CS. For accessing and updating their stored data for various application purposes, they may also dynamically interact with the CS. To save the computation resource as well as the online burden, cloud users may resort to TPA. In order to ensure the storage integrity of their outsourced data. Hoping to keep their data private from TPA.

Assuming that the data integrity threats towards user data can come from both internal and external attacks at CS. These may include: software bugs, hardware failures, bugs in the network path, hackers, malicious or accidental management errors, etc. Besides, CS can be self interested for their own benefits for activities such as to maintain reputation.It may also try to hide the data corruption incidents from users. The use of TPA service provides a cost-effective method for users to gain trust in cloud. The TPA, who is in the business of auditing, is considered to be reliable and independent. However, if the TPA could learn the outsourced data after the audit it would harm the User.

Beyond users' reluctance to leak data to TPA, it is also assumed that cloud servers have no incentives to reveal their hosted data to the external or other parties .Where as on the one hand, there are some regulations, e.g., HIPAA [8], requesting CS to maintain the privacy of users data. On the other hand, as users' data belong to their business asset [5], there also exist financial incentives for CS to protect it from any external parties. Therefore neither CS nor TPA has motivations to collide with each other during the auditing process. To Authorize the CS to respond to the audit delegated to TPA's, the user can issue a certificate on TPA's public key, and all audits from the TPA are authenticated against such a certificate.

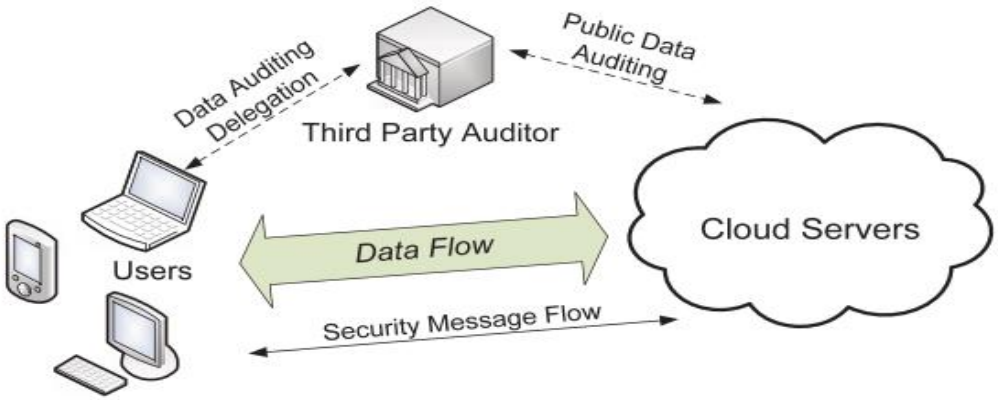

Figure1: System Architecture

4. DESIGN GOALS

To enable privacy-preserving public auditing for cloud data storage under the aforementioned model. The protocol design should achieve the following security and performance guarantees.

1. Public audit ability: To allow TPA to verify the correctness of the cloud data on demand without retrieving a copy of the whole data or introducing additional online burden to the cloud users.
2. Storage correctness: To ensure that there exists no cheating cloud server that can pass the TPA's audit without indeed storing users' data intact.
3. Privacy-preserving: To ensure that the TPA cannot derive users' data content from the information collected during the auditing process.
4. Lightweight: To allow TPA to perform auditing with minimum communication and computation overhead.
5. Batch auditing: To enable TPA with secure and efficient auditing capability to cope with multiple auditing delegations from possibly large no. of different users simultaneously [1].

5. PROPOSED SCHEMES

This section presents public auditing scheme which provides a complete outsourcing solution of data and its integrity checking.

The public auditability is a main drawback of cloud computing technology. Secure public auditing scheme for cloud storage provide more security compared previous technology. Which present the main result for privacy preserving Public auditing to achieve the before mentioned design Goals. Finally, it show how to extent the main scheme to batch auditing and encryption algorithms. The batch Auditing used to audit the group of details. The proposed problem is multi write and problem of TPA if Third-party-auditor not only uses data but also modify the data than how data owner or user will know about this problem. Here the user has two types' keys, one of which only the owner knows called private key and another one which is known to anyone called public key. Match both the data it must be same as the sent one on the sender cannot deny that they sent it. The downloading of data for its integrity verification is not feasible task since it's very costly because of the transmission cost across the network.

A. Public Auditing:

Public auditing scheme algorithms are 1. KeyGen, 2.SigGen, 3.GenProof 4.Verify Proof. *KeyGen* is a key generation algorithm that is run by the user to setup the scheme. *SigGen* is used by the user to generate verification Meta data. *Gen Proof* is run by the cloud server to generate a proof of data storage correctness. *VerifyProof* is run by the TPA to audit the proof from the cloud server.

B. Batch Auditing:

Secure privacy-preserving public auditing in Cloud Computing, TPA may concurrently handle multiple Auditing delegations upon different users' requests. The individual auditing of these tasks for TPA can be tedious and very inefficient. Given *A* auditing delegations on *A* distinct data files from *A* different users, it is more advantageous for TPA to batch these multiple tasks together and audit at one time.

C. Access Control:

Access control mechanisms are tools to ensure authorized user can access and to prevent unauthorized access to information systems. The following are six control statements should be consider ensuring proper access control management as in

1) The Access to information.
2) Manage user access rights.
3) Encourage good access practices.
4) Control access to the operating systems.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 3, Issue 1, pp: (141-146), Month: January - March 2015, Available at: www.researchpublish.com

5) Control access to network services.

6) Control access to applications and systems.

The proposed problem can be generalized as how can the client find an efficient way to perform periodical integrity verifications without the local copy of data files, as in. If any two users or more users are using a data, one is writing a data while one is reading a data than it may be wrong read by 1 user, so to resolve data inconsistency is become an important task of the data owner and another problem how to trust on TPA is not calculated. If TPA become intruder and pass information of data or deleting a data than how owner know about this problem are not solved. Integrity and consistency is a must.

6. DEFINITIONS AND FRAMEWORK

A public auditing scheme consists of four algorithms (KeyGen, SigGen, GenProof, and VerifyProof). KeyGen is a key generation algorithm that is run by the user to setup the scheme. SigGen is used by the user to generate verification metadata, which may consist of MAC, signatures, or other related information that will be used for auditing. GenProof is run by the cloud server to generate a proof of data storage correctness, while VerifyProof is run by the TPA to audit the proof from the cloud server.

Running a public auditing system consists of two phases, Setup and Audit:

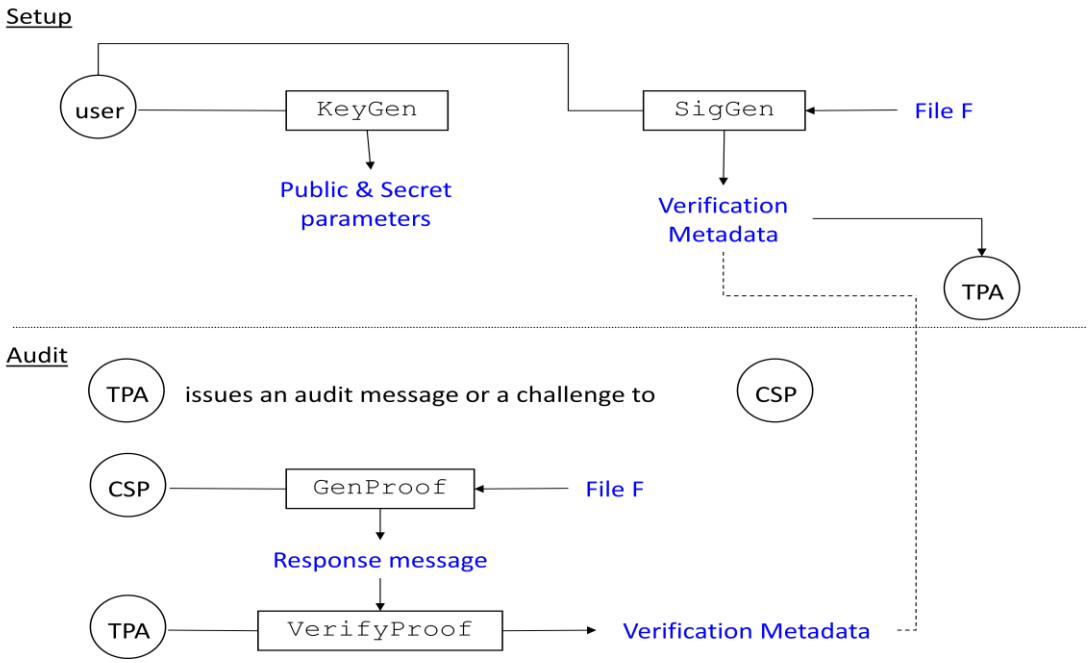

Figure 2- Public Auditing Schemes

1. Setup: The user initializes the public and secret parameters of the system by executing KeyGen, and preprocesses the data file F by using SigGen to generate the verification metadata. The user then stores the data file F and the verification metadata at the cloud server, and delete its local copy. As part of pre-processing, the user may alter; the data file F by expanding it or including additional metadata to be stored at server.

2. Audit: The TPA issues an audit message or challenge to the cloud server to make sure that the cloud server has retained the data file F properly at the time of the audit. The cloud server will derive a response message from a function of the stored data file F and its verification metadata by executing GenProof. The TPA then verifies the response via VerifyProof. Framework assumes the TPA is stateless, which is a desirable property achieved by our proposed solution.

The TPA is stateless, i.e., TPA does not need to maintain and update state between audits, which is a desirable property in the public auditing scheme [4].

7. ALGORITHM

Like DES, AES is a symmetric block cipher. This means that it uses the same key for both encryption and decryption. However, AES is quite different from DES in a number of ways. The algorithm Rijndael allows for a variety of block and key sizes and not just the 64 and 56 bits of DES' block and key size. The block and key can in fact be chosen independently from 128, 160, 192, 224, 256 bits and need not be the same. However, the AES standard states that the algorithm can only accept a block size of 128 bits and a choice of three keys - 128, 192, 256 bits. Depending on which version is used, the name of the standard is modified to AES-128, AES-192 or AES256 respectively. As well as these differences AES differs from DES in that it is not a feistel structure. Recall that in a feistel structure, half of the data block is used to modify the other half of the data block and then the halves are swapped. In this case the entire data block is processed in parallel during each round using substitutions and permutations.

A number of AES parameters depend on the key length. For example, if the key size used is 128 then the number of rounds is 10 whereas it is 12 and 14 for 192 and 256 bits respectively. At present the most common key size likely to be used is the 128 bit key. This description of the AES algorithm therefore describes this particular implementation.

Rijndael was designed to have the following characteristics:

• Resistance against all known attacks.

• Speed and code compactness on a wide range of platforms.

• Design Simplicity.

The input is a single 128 bit block both for decryption and encryption and is known as the **in** matrix. This block is copied into a **state** array which is modified at each stage of the algorithm and then copied to an output matrix. Both the plaintext and key are depicted as a 128 bit square matrix of bytes. This key is then expanded into an array of key schedule words (the **w** matrix). It must be noted that the ordering of bytes within the **in** matrix is by column. The same applies to the **w** matrix.

Inner Workings of a Round:

The algorithm begins with an **Add round key** stage followed by 9 rounds of four stages and a tenth round of three stages. This applies for both encryption and decryption with the exception that each stage of a round the decryption algorithm is the inverse of its counterpart in the encryption algorithm. The four stages are as follows:

1. Substitute bytes

2. Shift rows

3. Mix Columns

4. Add Round Key

The tenth round simply leaves out the **Mix Columns** stage. The first nine rounds of the decryption algorithm consist of the following:

1. Inverse Shift rows

2. Inverse Substitute bytes

3. Inverse Add Round Key

4. Inverse Mix Columns

Again, the tenth round simply leaves out the **Inverse Mix Columns** stage.

8. CONCLUSION

Here proposed is a privacy-preserving public auditing system for data storage security in Cloud Computing. The AES and random masking guarantees 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 burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage.

9. FUTURE SCOPE

With the establishment of privacy-preserving public auditing in cloud computing, TPA may concurrently handle multiple auditing delegations upon different user's requests. The individual auditing of these tasks for TPA can be tedious and inefficient. Batch auditing not only allows TPA to perform the multiple auditing tasks simultaneously, but also reduces the computation cost on TPA side. And also we can extend our work to support for the data dynamics which includes the block level operations of modification, deletion, insertion.

ACKNOWLEDGEMENTS

This work is supported by Department of Computer Engineering NDMVPS's KBTCOE Nashik for providing all necessary facilities and their support. We also thanks to Prof. R. R. Tajanpure for guiding us to understand the work conceptually and also for her constant and encouragement to complete this work.

REFERENCES

[1] Cong Wang, Member, IEEE, Sherman S.M. Chow, Qian Wang, Member, IEEE, Kui Ren, Senior Member, 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. ACM Workshop Cloud Computing Security (CCSW 09).pp. 43-54, 2009

[5] M.A. Shah, R. Swaminathan, and M. Baker,"Privacy-Preserving Audit and Extraction of Digital Contents," Cryptology ePrint Archive, Report 2008/186, 2008.

[6] C. Wang, K. Ren, W. Lou, and J. Li, "Towards Publicly Auditable Secure Cloud Data Storage Services," IEEE Network Magazine, vol. 24, no. 4, pp. 19-24, July/Aug. 2010.

[7] 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.

[8] 104th United States Congress, "Health Insurance Portability and Accountability Act of 1996 (HIPPA)," http:// aspe.hhs.gov/ admnsimp/pl104191.htm, 1996.