International Journal of Innovative Research in Computer Science & Technology (IJIRCST)
ISSN: 2347-5552, Volume-5, Issue-3, May 2017
DOI: 10.21276/ijircst.2017.5.3.3

Review of Data Integrity Checking in Cloud Computing

HanumanthaRao.Galli, Padmanabham.P

Abstract— **Cloud computing is the emerging era in many fields of computing. Here the major role is that storing of user data and data has to provide to the users whenever they needed. There are many challenges will takes place to perform store and providing of information to the user. One of the challenges in cloud computing is the integrity of the data stored in cloud. Here we are identifying existing techniques which are used for data integrity checking in cloud computing.**

Index Terms— **Cloud computing, Integrity.**

I. INTRODUCTION

With the development of technology and communication, the data has been placed important role in the computing world. People begin start on to use cloud storage area to place their sensitive information. And IT projects store their information in a flexible on-demand manner so that burden will be less for storage as well as maintenance and costs on software and hardware and provide convenience to the outsourced data.

In the public cloud model, the clients store their large data in the public cloud. Although cloud provides many advantages and flexibility for the clients, it faces many security challenges. Since the data stored in cloud is in the outside and the user no longer possesses their data locally, it entails the security problems in terms of confidentiality, integrity and availability of data and services. To make sure the security of the outsourced data, the user desires to occasionally check data integrity in order to be convinced that data are correctly stored in the cloud. For the data owners, the main confront is how to perform occasionally integrity checking without the local copy of data files.

II. PROCEDURE FOR PAPER SUBMISSION

A. *Review Stage*

Ms. T.J.SALMA [1], G. Ateniese et al. [2] are the first to consider public auditability in their defined "provable data possession" (PDP) model for ensuring possession of data files on untrusted storages.

Manuscript Received May 6, 2017

HanumanthaRao.Galli, Research Scholar, Department of Computer Science and Engineering, JNTUH, Hyderbad, INDIA,

Padmanabham.P, Director, Department of Computer Science and Engineering, JNTUH, Ranga Reddy, INDIA,

This scheme utilizes the RSAbased homomorphic authenticators for auditing
outsourced data and suggests randomly sampling a few blocks of the file.However, the public auditability in this scheme demands the linear combination of sampled blocks exposed to external auditor. When used directly, their protocol is not provably privacy preserving, and thus may leak user data information to the auditor.

M. A. Shah et al. [3] propose allowing a TPA to keep online storage honest by first encrypting the data then sending a number of pre-computed symmetric-keyed hashes over the encrypted data to the auditor. The auditor verifies both the integrity of the data file and the server's possession of a previously committed decryption key. This scheme only works for encrypted files and it suffers from the auditor statefulness and bounded usage, which may potentially bring in on-line burden to users when the keyed
hashes are used up.

Q. Wang et al. [4] propose to combine BLS based homomorphic authenticator with MHT (Merkle Hash Tree) to support both public auditability and fully data dynamics, simultaneously.

Prof. Hitesh Patel, Prof. Parin Patel, Prof Kiran Patel[5], Here to achieve integrity used BLACK hash function for generating hash code but it gives lightweight data integrity verification which is applicable small amount of data sizes.

Qian Wang and Kui Ren Wenjing Lou Yanchao Zhang[6], To ensure the integrity of data shares, an efficient dynamic data integrity checking scheme is constructed based on the principle of algebraic signatures. we show that our scheme is highly secure and efficient, thus can be implemented in the current generation of sensor networks.

S. P. Jaikar,M. V. Nimbalkar[7], We have analyzed the data security concerns in cloud data storage, which is a distributed storage system. We proposed a distributed scheme to ensure users that their data are indeed stored appropriately and kept intact all the time in the cloud. To provide redundancy we used erasure correcting code in the file distribution preparation. As we all know cloud is not just a third party data warehouse. So providing support for dynamic operations is very important. Our scheme maintains the same level of storage correctness assurance even if users modify, delete or append their data files in the cloud. Challenge response protocol along with pre-computed token is used to verify the storage correctness of user"s data & to effectively locate the malfunctioning server when data corruption has been detected. Through detailed performance analysis, we show that our scheme is

having very low communication overhead & guarantees to detect every single unauthorized data modification. Our scheme has no limitation on number of pre-computed tokens used for challenging the cloud servers. Unlimited number of challenges can be made. We removed burden of calculating pre-computed tokens & storing the locally from the users. By splitting the file according to the number of server ' s we are added extra security to system. But we still believe that data storage security in Cloud computing is an area full of challenges and of paramount importance.

B. Original Provable Data Possession

In this method, the data is pre-processed before sending it to the cloud server. Here the data is filled with some tag value to verify at the client side. Complete data is sent over to the server and at the client side meta-data is stored. This meta-data is used for the verification as per user need. To check the integrity user will sends the challenge to the server at that time server will respond with the data. Then the client will compare the reply data with the local meta-data. In this way client will check that the data is modified or not. Original PDP has low computation and storage overhead. It supports both encrypted data and plain data. It provides public verifiability. It is efficient because small portion of the file needs to be accessed to generate proof on the server. This technique is only applicable to the static files. Homomorphic hashing technique is employed to compose multiple block inputs into a single value to reduce the size of proof.

C. Scalable PDP

Scalable PDP is an extended version of the original PDP. Original PDP uses public key to reduce computation overhead whereas Scalable PDP uses the symmetric encryption. Scalable PDP provides dynamic operation on remote data. Bulk encryption is not required by scalable PDP. It uses the symmetric-Key which is more efficient than public-Key encryption.

D. Dynamic PDP

Dynamic PDP supports full dynamic operations like insert, update, modify, delete etc. In this technique the dynamic operation permits the authenticated insert and delete functions with rank-based authenticated directories and with a skip list. Though DPDP has some procedure quality it is still economical. For example, to generate the proof for 1GB file, DPDP only produces 415KB proof data and 30ms procedure overhead. It provides comparatively greater computational, communication, and storage overhead.

E. Message Authentication Code (MAC) method

The outsourced data file F consists of a finite ordered set of blocks m1; m2; . . . mn. One simple way to ensure the data integrity is to pre-compute MACs for the complete file. Before data outsourcing, the data owner pre-computes MACs of F with a set of secret keys and stores them locally. During the auditing process each time , the data owner reveals a secret key to the cloud server and asks for a fresh keyed MAC to verify it. This method provides deterministic data integrity assurance because the verification covers all the data blocks. However, the number of verifications can be performed in this solution is limited by the number of secret keys. Once the keys are exhausted, the data owner has to retrieve the entire file of F from the server in order to estimate new MACs, which is usually impractical due to the large communication overhead. Private keys are required for verification so public audit ability is not supported.

F. Signature Method

The data owner pre-computes the signature of each block and sends both F and the signatures to the cloud server for storage. Data owner can adopt a spot-checking approach to verify correctness of F. i.e., requesting a number of randomly selected blocks and their corresponding signatures to be returned.

Note that above methods can only support the static data and also a large communication overhead that greatly affects system efficiency [8].

III. PRINICIPAL OF DATA INTEGRITY

A. System Model

Cloud storage applications offer client (data owner) the opportunity to store, backup or archive their data in the cloud storage network. Such applications should ensure data integrity and availability on a long term basis[9]. This objective requires developing appropriate remote data possession verification. Representative network architecture for cloud data storage is illustrated in Fig. 1. As shown in figure three different network entities can be identified as follows:

Client: Users, who have data to be stored in the cloud and rely on the cloud for data computation, consist of both individual consumers and organizations. Cloud Storage server: is managed by the cloud service provider (CSP) to provide data storage service and has significant storage space and computation resources.

Third Party Auditor (TPA): an optional TPA, who has expertise and capabilities that users may not have, is trusted to assess and expose risk of cloud storage services on behalf of the users open request.

Verifier may be User (Data owner) or third party auditor. The role of the verifier fall into two categories:

Private Auditability: It allows only data owner for checking the integrity of the data file stored on cloud server.

Public Auditability: It allows anyone, not just the client (data owner), to challenge the cloud server for correctness of data.

B. Threat Model

We consider the third party auditor is honest-but-curious. It performs honestly during the whole auditing procedure but it is curious about the received data. Thus, for the storage of secured data, there is also a privacy requirement for the third party auditing protocol. That is, no data will be leaked out to the third party auditor during the auditing procedure. But the server is dishonest and may conduct the following attacks:

Replace Attack: Suppose the Server discarded a challenged data block mi or its metadata ti, in order to pass the auditing, it may choose another valid and uncorrupted pair of data block and metadata (mk, tk) to replace the

original challenged pair of data block and metadata (mk, tk).

Replay Attack: The Server generates the proof from the previous proof or other information, without querying the actual Owner's data.

Forge Attack: The Server may forge the metadata of data block and deceive the auditor.

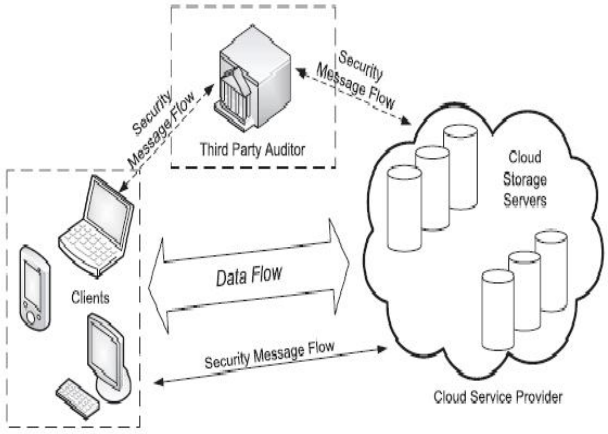

Fig 1: Cloud Data Storage Architecture

IV. CONCLUSION

In this paper we observed that data integrity is emerging area in cloud computing for security purpose. Researcher proposed efficient new techniques based on the PDP and PoR schemes. PDP scheme easily support dynamic operation but it doesn't include error correcting code. so significant amount of overhead in the PoR scheme comes from the error-correcting codes which are not present in the PDP scheme. Therefore we can say that designing efficient, secure and fully dynamic remote data integrity is still open area of research.

REFERENCES

[1] Ms. T.J.SALMA, A Flexible Distributed Storage Integrity Auditing Mechanism in Cloud Computing
[2] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D.Song, "Provable data possession at untrusted stores," Cryptology ePrint Archive, Report2007/202, 2007, http://eprint.iacr.org/.
[3] M. A. Shah, R. Swaminathan, and M. Baker, "Privacy-preserving audit and extraction of digital contents," Cryptology ePrint Archive, Report 2008/186,2008,http://eprint.iacr.org/.
[4] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in Proc. of ESORICS'09,Saint Malo, France, Sep. 2009.
[5] Prof. Hitesh Patel, Prof. Parin Patel, Prof Kiran Patel, "Achieving Data Integrity in Cloud Storage Using BLAKE Hash Function ",2014 IJEDR | Volume 2, Issue 2 | ISSN: 2321-9939
[6] Qian Wang and Kui Ren Wenjing Lou Yanchao Zhang," Dependable and Secure Sensor Data Storage with Dynamic Integrity Assurance", IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2009 proceedings 954-962.
[7] S. P. Jaikar,M. V. Nimbalkar," Verifying Data Integrity in Cloud ", International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 3– No.1, July2012 – www.ijais.org 38-46
[8] Ms.RohiniG.Khalkar and Prof. Dr. S.H.Patil," DATA INTEGRITY PROOF TECHNIQUES IN CLOUD STORAGE", International Journal of Computer Engineering and Technology (IJCET), ISSN 0976-6367(Print), ISSN 0976 – 6375(Online) Volume 4, Issue 2, March – April (2013), © IAEMEpp. 454-458
[9] T S Khatri , Prof G B Jethava," Survey on data Integrity Approaches used in the Cloud Computing", International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 9, November – 2012 ISSN: 2278-0181 1-6.

