EAI Endorsed Transactions
on Cloud Systems

Research Article **EAI.EU**

A Prototype for Data Integrity in Cloud Environment

Shashi[1,*], Suryakant Yadav[2] and Anuranjan Mishra[3]

[1]Research scholar, Noida International University, Gr. Noida, India
[2]Noida International University, Gr. Noida, India
[3]GNIOT, Gr Noida, India

Abstract

Currently in world wide computing (Cloud) has a great impact on life. Everyone can access the all services of cloud if he/she is on different location. Client user can access the cloud services as per their requirement. If one user on cloud, Integrity of data is an important aspect. Data integrity is the upkeep of and the confirmation of the exactness and consistency of, data over its whole life-cycle, and is a basic perspective to the outline, execution and use of any framework which stores, forms, or recovers data. In this research paper, a fitting technique that guarantees the integrity of data and in addition rightness of calculations done by the cloud service provider is introduced.

Integrity is a method for protecting the consistency of the put away data in cloud server and guaranteeing the innovation of the data put away in the cloud server. It implies that the data can be altered just by approved people, along these lines expanding the certification, confirmation and dependability of the cloud service providers.

Keywords: Data Integrity, Authentication, Storage, Data.

Received on 31 May 2020, accepted on 14 August 2020, published on 07 September 2020

doi: 10.4108/eai.7-9-2020.166287

1. Introduction

In this, the integrity of the outsourced data put away in the un-confided in remote cloud servers has been guaranteed. It has been finished by executing a strategy that gains a proof of data ownership by creating metadata of the data in the In this, the integrity of the outsourced data put away in the un-confided in remote cloud servers has been guaranteed. It has been finished by executing a strategy that gains a proof of data ownership by creating metadata of the data in the cloud. This evidence checks that the data put away in the remote cloud server are not changed by unapproved clients, along these lines guaranteeing the data integrity. Along these lines, this verification protocol keeps the remote cloud stockpiling servers and unapproved people from harming, distorting or changing the data without the learning of the data proprietor by directing incessant security minds the data stockpiling. This evidence confirms that the data put away in the remote cloud server are not adjusted by unapproved clients, accordingly guaranteeing the data integrity. Along these lines, this verification protocol keeps the remote cloud stockpiling servers and unapproved people from harming; distorting or changing the data without the information of the data proprietor by leading regular security minds the data storage.

2. Integrity Checking Methods

2.1 Message Authentication Code (MAC):

To confirm the data integrity, MAC for the whole data is created by the DO before putting away the data record in a remote server. It is held by the DO in the neighborhood stockpiling, however the first data is put away in the remote server. Keeping in mind the end goal to confirm the integrity of the data, the data proprietor recovers the whole data from the remote server, re-figures the MAC

[*]Corresponding author. Email: teotia.shashi@gmail.com

and contrasts it and the one that is held in the nearby capacity.

2.2 Checksum:

Before storing the data file in the cloud server, the data user computes the checksum [1] of the data file and retains the checksum in the local storage. To verify the data file, the data owner retrieves the data from the server and computes the checksum. If this checksum matches with the checksum in the local storage, the data user ensures the integrity of the data file.

2.3 Provable Data Possession (PDP):

The model of Provable Data Possession scheme [2] assures the integrity of data files stored in the cloud servers. This scheme uses RSA based homomorphic tags for verifying the data integrity. These tags should be precomputed by the data owner and later used for verification. The drawback of this scheme is a computation and storage overhead in generating and maintaining the tokens.

2.4 Pre-computed tokens:

Before storing the data in the server, a number of verification tokens [3] each covering a portion of data blocks are computed and retained by the data owner in the local storage. To verify the correctness of the data stored in the cloud server, the cloud user challenges the server indicating the position of data blocks. For that, the cloud server generates the signature on the indicated data blocks and sends it to the cloud user.

2.5 Proof of Retrievability (POR):

This technique [4] utilizes some sentinel characters to check the integrity of the data document put away in the cloud server. These sentinel characters are tucked away among the first data hinders by the data clients. The places of the sentinel characters are kept up in the neighborhood stockpiling by the data proprietor. For checking the integrity, the data proprietor sends the test message indicating the position. The server reacts with the relating character in the asked for position. At that point the server settles on the integrity of the data hinder by coordinating the characters.

3. Model for Data Integrity Assurance:

This system model for secured storage consists of various entities like Data Owners (DO), Data Users (DU), Cloud Service Providers (CSP) comprising of n cloud storage servers (s1, s2,.. sn) and a number of compute intensive servers, a separate service for generating meta-data and a Third Party Auditor (TPA). The system model for the secured storage is represented in the Figure 1.1.

3.1 Third Party Auditor (TPA):

It refers to an organization or an individual who is having the capability to verify the integrity of the stored data in cloud server.

The role of the auditor falls into two categories.

1. **Private auditability:** In this, only the data owner is permitted to check the integrity of the data file residing on the server.

2. **Public auditability:** It means that anyone, including the TPA is permitted to verify the data integrity.

3.2 Metadata generator:

It is an entity or a service that is actually running on the premise of cloud service providers, but under the control of the Data Owners. It is performing the task of generating the metadata from the encrypted data blocks.

4. Assuring the Integrity of Data and Computation

For ensuring the correctness of user's data stored in the cloud server, an effective scheme has been proposed here. This scheme benefits with two salient features.

- Assuring integrity of the data stored in cloud server by obtaining a proof that the data is not modified or deleted by the cloud service provider without the consent of the data owner.
- Ensuring secure computation by allowing the data users to verify the correctness of computations done by the cloud service provider, thereby avoiding the misbehavior of the server.

4.1 Stored data integrity assurance

In this proposed scheme, the integrity of the stored data has been ensured by the DO or by the TPA by using the following sequence of operations.

5. Metadata generation phase

1. The data file is split into nb data blocks by the data owner.
2. All the data blocks have been encrypted using the 2-Keys symmetric encryption algorithm by the encryption service,

thereby producing the encrypted data blocks EDBs.

3. Then the EDBs are given as input to the metadata generator.
4. The metadata generator applies some random function RF and the secret key Sk, received from the data owner over the encrypted data blocks EDBs and generate metadata blocks MDBs.
5. These MDBs are attached at the end of the encrypted data blocks EDBs by the metadata generator.
6. The combination of EDBs and MDBs are stored together as a single data file in the cloud server by the metadata generator.

6. Verification phase

1. The data owner submits the random function and the secret key to the third party auditor.
2. For verifying the data integrity, the TPA issues a challenge (i, j) message to the cloud service provider.
3. The CSP responds with the two characters of EDB [i, j] and the corresponding MDB [i+n, j] to the TPA.
4. Upon receiving the 2 characters, the TPA applies the inverse random function and secret key for the metadata character and generates the character C'.
5. If this character C' matches with the EDB [i, j], then the TPA concludes that the data has not been corrupted.

The architecture for illustrating the integrity verification of data stored in the server is depicted in the Figure 1.2.

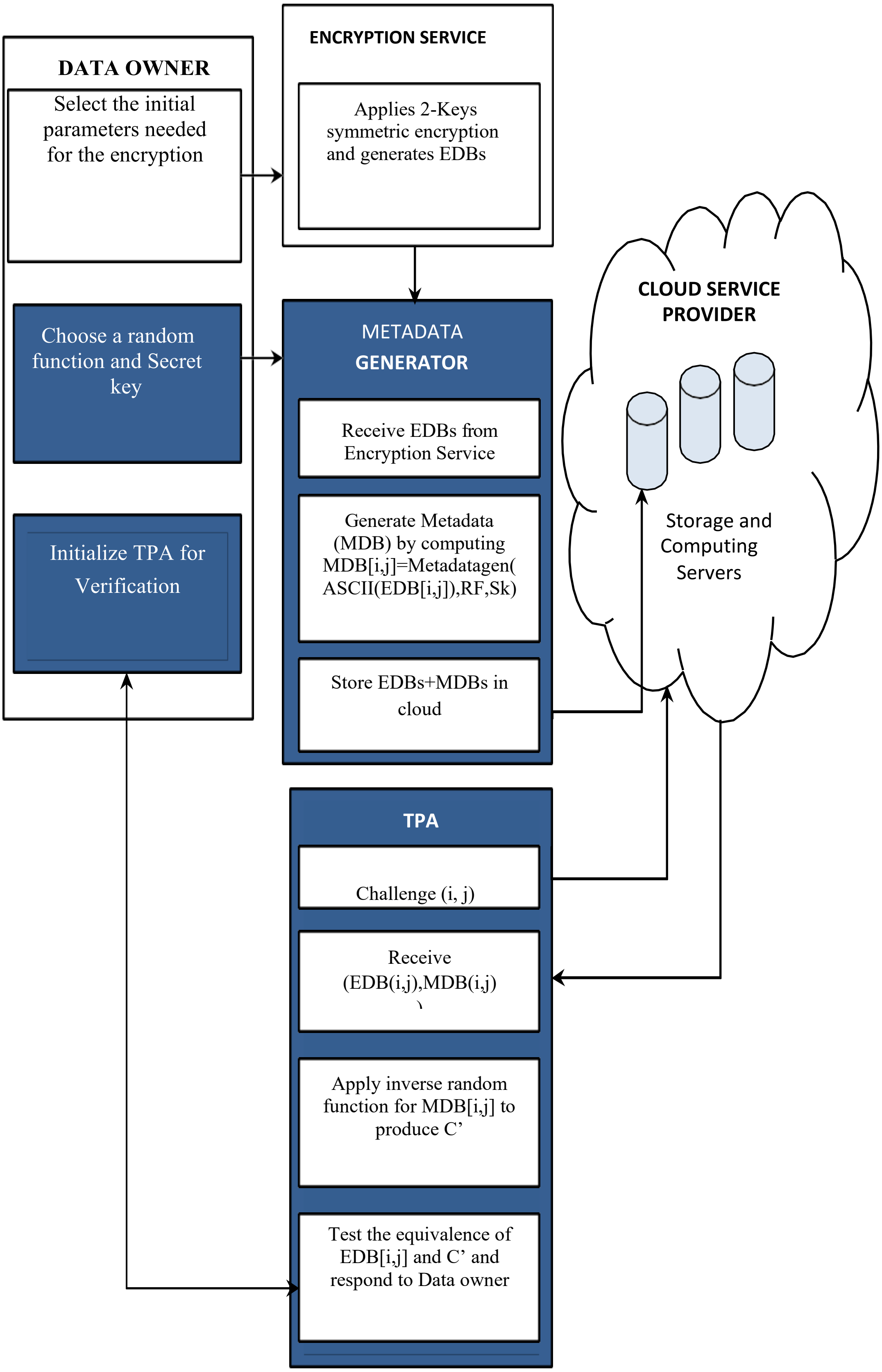
DATA OWNER
Select the initial parameters needed for the encryption
Choose a random function and Secret key
Initialize TPA for Verification
ENCRYPTION SERVICE
Applies 2-Keys symmetric encryption and generates EDBs
METADATA GENERATOR
Receive EDBs from Encryption Service
Generate Metadata (MDB) by computing MDB[i,j]=Metadatagen(ASCII(EDB[i,j]),RF,Sk)
Store EDBs+MDBs in cloud
CLOUD SERVICE PROVIDER
Storage and Computing Servers
TPA
Challenge (i, j)
Receive (EDB(i,j),MDB(i,j)
Apply inverse random function for MDB[i,j] to produce C'
Test the equivalence of EDB[i,j] and C' and respond to Data owner

Figure 1.2. Architecture for data integrity in cloud environment

This data integrity verification system prevents the malicious users or the un-trusted servers from either distorting or modifying the information maintained in it without the permission of the data owner by conducting frequent integrity checks over the data blocks. For accomplishing this, metadata is generated for each data block and it is appended to the original data block. Then this entire data block is stored in the cloud storage servers. In order to verify the integrity of the data blocks, the data owner sends the initTPA() message to the TPA. Upon receiving this initTPA() message, the TPA throws a challenge(i,j) message to the cloud server. The cloud server responds with the two characters to the TPA (i) the character at EDB[i,j] and (ii) the character at MDB[i+nb, j]. The TPA applies the inverse random function and secret key to the character at MDB[i+nb, j] and produces a character C'. If the value of C' and EDB[i,j] are the same, then the TPA confirms the integrity of the stored data and sends a positive reply to the data owner. The entire operations can be separated into two phases, namely metadata generation phase and verification phase.

7. Conclusion

In this paper, we design and deployment of integrity assurance model that ensures the integrity of data stored in the cloud server and correctness of computations done by the cloud service provider is presented. We also study the metadata generation, data integrity verification, computation commitment generation and computation verification and architecture also that show how data is stored on server. In our next study, we produce the algorithm for data integrity and obtained the computation of data that is stored in the cloud are not modified by the provider or a third-party individual, thereby ensuring the integrity of data by generating meta-data for the data blocks stored in cloud server.

References

[1] W. Stallings, “Network security essentials: applications and standards”, Third edition, ISBN-9788131716649, Prentice hall, 2007

[2] 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, New York, NY, USA, pp. 598– 609, 2007.

[3] Francesc Sebffie, Josep Domingo-Ferrer, Antoni Martinez-Balleste, Yves Deswarte, and Jean-Jacques Quisquater, “Efficient remote data possession checking in critical information infrastructures”, IEEE Transactions on Knowledge and Data Engineering, 2008.

[4] Juels, A. and B.S. Kaliski, “PORs: Proofs of Retrievability for large files”, Proceedings of the 14th ACM Conference on Computer and Communications Security, (CCS’ 07), ACM Press, New York, pp. 584- 597, 2007.

[5] Peter Mell, Timothy Grance, “The NIST definition of cloud computing”, NIST Special Publication 800-145, National Institute of Standards and Technology, Information Technology Laboratory, September 2011.

[6] Mazhar Ali, Samee U. Khan and Athanasios V. Vasilakos, ”Security in cloud computing:opportunites and challenges”, Information Sciences, 305, pp. 357-383, 2015.

[7] Mahesh S.Giri, Bhupesh Gaur, Deepak Tomar , “A Survey on data integrity techniques in cloud computing“, International Journal of Computer Applications, Volume 122, pp. 27-32, July 2015.

[8] Min Xie, HaixunWang, Jian Yin, and Xiaofeng Meng, “Integrity auditing of outsourced data”, in VLDB '07: Proceedings of the 33rd International Conference on Very Large Databases, pp. 782-793, 2007.

[9] Osama Harfoushi, Bader Alfawwaz, Nazeeh A. Ghatasheh, Ruba Obiedat, Mua’ad M. Abu-Faraj, Hossam Faris, “Data security issues and challenges in cloud computing: A conceptual analysis and review”, Communications and Network, 6, pp.15-21, 2014.

[10] Dimitrios Zissis and Dimitrios Lekkas, “Addressing cloud computing security issues”, Future Generation Computer Systems 28, pp. 583– 592, 2012.