Preserving the Integrity of Big Data in Cloud: A Survey

E. Angelin Kanimozhi
Department of Computer Science and Engineering
Thiagarajar College of Engineering (TCE)
Madurai, India
angelinkanimozhi@gmail.com

M.Suguna
Department of Computer Science and Engineering
Thiagarajar College of Engineering (TCE)
Madurai, India
mscse@tce.edu

Dr.S.Mercy Shalini
Department of Computer Science and Engineering
Thiagarajar College of Engineering (TCE)
Madurai, India
hodcse@tce.edu

***Abstract*—Big data can be defined as extremely large datasets, which can be structured or unstructured, and difficult to process. The biggest challenge nowadays, is to store and manage big data. There are many technologies available for storing the big data. Storage infrastructures like DAS and NAS were initially used for storing big data. Data analytics solutions like Hadoop, Hive, NoSQL etc. played a major role in big data storage. Cloud computing technology used remote servers to store big data. On storing big data in a cloud, it is important to preserve the data integrity. Data integrity is the property of the data to remain unmodified in the storage location. The cloud users may be concerned about the integrity of their data. Hence the cloud authority provides some methods like Provable Data Possession (PDP) protocols, to guarantee their users about the safety of their data in the cloud. This paper compares several frameworks that have been proposed so far, to ensure the integrity of the big data in a cloud. The methodologies discussed in those frameworks and their limitations are analyzed in this paper. In the end, a new idea is proposed to address the same problem.**

***Keywords*—Cloud computing, data integrity, Provable Data Possession (PDP), blockless verification, batch auditing, Remote Data Possession Checking (RDPC), data dynamics.**

I. INTRODUCTION

In this digital universe, we are seeing information explosion every day. The technical people who could work in data centers are in huge demand nowadays. This is because of the need of heavy technologies to store the overwhelming amount of information generated by the world, each and every day. Since this information is voluminous and heterogeneous, we call it as 'Big Data'. Generally, we define big data, in terms of five V's. They are velocity, volume, variety, value and veracity.

Velocity deals with the speed of transmission and the access to the data. There are billions of data are generated in the internet which has to be transmitted, stored and analyzed very quickly. Volume of a big data is nothing but the size of the immensely vast amount of data generated by social media, sensors, cell phones, etc. It is clearly an engineering challenge to determine the storage location for storing this incredible amount of data. The value of big data is the worth of the extracted data. Since we spend so much on storing, analyzing and retrieving them, these must have some value and purpose. Variety denotes the heterogeneity of the big data. The data can be structured, semi structured or unstructured. Nowadays, the data generated by internet are mostly unstructured and of different types. The veracity of big data deals with the accuracy and the reliability of the data. Big data needs to be stored and analyzed in some way. Many technologies evolved to store this big data. One of such technologies, is cloud computing.

Cloud computing is a technology which enables its users to store their data in the remote servers and access them whenever they require. Millions of users can use a cloud to store their data; hence this leads to 'big data storage' in cloud. The users may be

concerned about the accuracy and safety of their data in those cloud servers.

Characteristics of cloud computing are, on-demand self service, broad network access, resource pooling, rapid elasticity, measured service and multi tenancy. On-demand self enables the provision of cloud resources on demand whenever they are required by the cloud users. The resources that are hosted in a private cloud network and are available for access from a wide range of devices, is termed ad broad network access.

Resource pooling offers the provision of resources to the customers and allows them to change their levels of service at will without being subject to any of the limitations of physical or virtual resources in cloud computing environment. Rapid elasticity refers to the ability to provide scalable services by the cloud server to the cloud users. Even when there is no specific interaction for a service change, that service change is still noted so that it can be dealt with a later date or negotiated completely. Such property is known as measured service. Multitenancy is one of the architectures in cloud computing where on top of primary software instances, one or more logical software instances are created and executed. Multitenancy is the backbone of cloud computing as it allows multiple users to work in a software environment at the same time, each with their own separate user interface, resources and services.

Integrity of the data is nothing but the property of the data to remain unmodified in the cloud servers. There are several methodologies used to assure the integrity of the data. In this survey, these frameworks that are used to verify the data integrity in cloud computing are compared and analyzed.

II. FEATURES OF DATA INTEGRITY CHECKING FRAMEWORKS

The frameworks that are designed for the purpose of integrity checking are desired to possess some properties. Some of such features of data integrity checking protocols are described below.

A. Usage of TPA

A trusted third party auditor (TPA) can be used for the purpose of integrity verification. The tags like MAC and hash values are generated in the user side and the sever side. The user here refers to the cloud users or the data owners and the server refers to the cloud server where actually the data resides. The TPA's work is to compare both the tags and generate a proof to the user that they are same.

B. Privacy preserving from TPA

Privacy preserving from TPA is that, the TPA must not able to read or decrypt the actual data while the process of verifying. The must remain in encrypted form throughout the integrity verification process.

C. Signature generation

If a user signs his/her data, someone else can verify the signature and can prove that the data originated from that user is unaltered. These digital signatures are mostly hash values.

D. Blockless verification

Integrity of the desired blocks can be verified by checking a single block which is the linear combination of all other blocks. This is known as blockless verification.

E. Public verifiability

Other than the data owner, anyone like third party auditor can verify the data integrity without even downloading the entire outsourced data.

F. Data Dynamics

The user must be able insert delete or modify the block whenever required dynamically. This property is stated as data dynamics.

G. Batch auditing

The term batch auditing refers to the ability of the verifier to perform a number of verification processes simultaneously.

H. Incentive provision

If the cloud service provider is found to be guilty, by failing the integrity test, it must give some incentive to the data owner as a compensative.

I. Data recovery

When a data is found to be corrupted or lost, the cloud server tries to recover the original data by using some algorithms.

.**Table 1: Data integrity checking frameworks: Analysis based on desired features.**

Frameworks	**Usage of TPA**	**Privacy Preserving**	**Signing**	**Blockless verification**	**Public verifiability**	**Data dynamics**	**Batch auditing**	**Incentive Provision**	**Data Recovery**
ID- based RDPC [1]	Yes	Yes	Yes	No	Yes	No	No	No	No
RDPC with enhanced security [2]	No	No	Yes	Yes	No	No	No	No	No
Indistinguishability obfuscation [3]	Yes	Yes	Yes	No	Yes	Yes	Yes	No	No
OOPDP [4]	Yes	Yes	Yes	No	Yes	Yes	No	No	No
Novel Efficient RDPC [5]	No	No	No	No	No	Yes	No	No	No
SEPDP [6]	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No
Privacy preserving CLPDP [7]	Yes	Yes	Yes	No	Yes	No	Yes	No	No
Public auditing with dynamic structure [8]	Yes	No	No	Yes	Yes	Yes	Yes	No	No
Incentive [9]	Yes	No	Yes	No	Yes	No	No	Yes	No
Erasure code-based [10]	No	N0	No	No	No	No	Yes	No	Yes

III. COMPARISON OF FRAMEWORKS ADDRESSING THE DESIRED FEATURES

There are several frameworks defined for verifying the integrity of the data residing in the cloud. These frameworks require some properties for integrity checking as mentioned in the previous section. These frameworks are compared, for the features they possess and their implementations

A. Frameworks using TPA

Yong Yu et al. [1] proposed a challenge response protocol as a two party agreement between TPA and the cloud server. The TPA challenges the cloud server with data blocks. The server replies with the response which is compared with tags generated by the users. The proof is generated and reported to the users by the TPA. Later, Yuan Zhang et al. [3] broke the assumption that the TPA has the capability to bear all verification costs. Hence they proposed a public verification scheme which puts lightweight computations on TPA and delegate most computations to the cloud.

In the scheme proposed by Yujue Wang et al. [4], the lightweight computations are done in online and heavy computations are done in offline in the user side. Hence the burden of TPA is reduced. Sanjeet Kumar Naya et al. [6] proposed a model where the computational overhead in TPA is reduced by eliminating pairing operations. Certificate generation phase is eliminated in [7] to reduce the computational overhead in TPA. Similarly, we can see the usage of TPA in [8] and [9] for verification purpose.

B. Frameworks addressing privacy preserving from TPA

In [1], Yong Yu et al. use KGC (key generation center) for encryption of the data in the user side. Only for this encrypted data, the tags will be generated by the user. Hence TPA will not be able to disclose the data. This is similar to the work of Debiao et al. [7]. Yuan Zhang et al. introduce an indistinguishability obfuscation method where the TPA only needs to verify the validity of the commitment generated by the cloud server. Hence the users' data need not be disclosed to the TPA. In [4], the encryption process is done in offline at user side before even reaching TPA. In the work of Sanjeet et al. [6] TPA fails to infer the data from the CSP's (cloud service provider) response. In addition to preserving the privacy of data, the identity privacy has been also preserved in [9].

C. Frameworks addressing signature generation phase

In [1], anyone with the access to signer's identity can verify the signature of the signer. Y. Yu et al. improves the algebraic signature algorithm by involving pseudo-random functions [2]. Yuan et al. executes a signing phase by constructing function encryption schemes for general circuits from indistinguishability obfuscation [3]. In [4], all the heavy computations in signing algorithm are done in offline phase. In [6], all the outsourced data are tagged with signature that is generated in the signature generation phase. Debiao He et al. propose a certificateless signature scheme [7]. In [9], the signature generation phase is essential because the data owner has to be identified at the time of incentive provision.

D. Frameworks addressing blockless verification

Yong Yu et al. states that the basic construction of an RDPC protocol has the no-block verification (blockless verification) feature [2]. Sanjeet et al. work defines that the blockless verification is used for reducing the bandwidth consumption [6]. Jian Shen et al. achieve blockless verifiability by using BLS-based homomorphic verifiable authenticator [8].

E. Frameworks addressing public verifiability

The work of Yong Yu et al. describes that the public verifiability enable anyone to audit the integrity of the outsourced data [1]. Yuan Zhang et al. suggest a public verification scheme with less overhead at the auditor's side by using indistinguishability obfuscation [3]. The semi-generic transformation proposed in [4] is applicable to any pubic verifiable PDP-related schemes. Sanjeet et al. states that with public auditability (verifiability), the data users can recourse the auditing task to a third party auditor [6]. In [7], a semi-trusted TPA can verify the integrity without downloading the data from the cloud server. Jian Shen et al. designed an efficient public auditing protocol with novel dynamic structure for the outsourced data in the cloud [8]. Huaqun Wang et al. propose a public auditing

scheme with the property of the anonymity of the cloud user [9].

F. Frameworks addressing data dynamics

In [3], Merkle hash tree technique is used to support data dynamics. Yujue Wang et al. states that Computational Diffie-Hellman (CDH) assumptions can be used for the dynamic PDP scheme [4]. Hao Yan et al. introduces a linear table called ORT (operational record table) for implementing data dynamics [5]. Debiao He et al. achieve data dynamics with lesser computation overhead [6]. In [8], a dynamic structure is designed by combining a doubly linked list information table and a location array for supporting data dynamics.

G. Frameworks addressing batch auditing

In Yuan Zhang et al. work, the batch verification overhead on the TPA side is independent of the number of verification tasks [3]. Sanjeet Kumar Nayak et al. work achieves batch auditing by aggregating many verification equations, requested by different data users, into one verification equation [6].In [7], the verifier can execute a large number of verification delegation simultaneously, thus achieving batch verification. In [8], the batch auditing is done for multiple files from one data owner and for multiple files from multiple data owners. The homomorphic integrity tags can be aggregated for batch verification in [10].

H. Frameworks addressing incentive provision and data recovery

Huaqun Wang et al. states that after the data integrity verification is completed, if the data is found to be corrupted or lost, the cloud organization must provide an incentive to the data owner, after proving their identity [9]. The new integrity tags are generated from the old integrity tags without the involvement of users' secret key or backup servers. This technique is nothing but the erasure code, used for data recovery purpose [10].

IV. Conclusion

In our paper, we have discussed about different parameters required for data integrity checking and about various frameworks designed for the same. The main drawback found in all these frameworks is, the data corruption is found only at the time of verification. A data integrity check happens only when the user demands. If the user doesn't demand, they will be kept unaware of the corruption or loss of their data in the servers. To overcome this, we have come up with the idea of introducing blockchain technology for the storage of data in cloud. The hash values of the cloud data will be stored in a common ledger called blockchain and shared among different servers. Hence the data corruption is identified at the time of occurrence and the corrupted data can be rolled back to the original state, as a part of data recovery.

References

[1] Yong Yu Man Ho Au, Giuseppe Ateniese, Xinyi Huang, Willy Susilo, Yuanshun Dai and Geyong Min, *"Identity-Based Remote data integrity checking with perfect data privacy preserving for cloud storage"*, IEEE Transactions on information forensics and security, vol. 12, no. 4, 2017.

[2] Yong Yua b, Yafang Zhang, Jianbing Ni a, Man Ho Auc, Lanxiang chend and Hongyu Liua,"*Remote data possession checking with enhanced security for cloud storage"*, Elsevier, Journal of Future Generation computer systems, vol. 52, pp. 77-85, 2015.

[3] Yuan Zhang, Chunxiang Xu, Xiaohui Liang, Hongwei Li, Yi Mu, and Xiaojun Zhang, "*Efficient public verification for cloud storage systems from indistinguishability obfuscation"*, IEEE Transactions On Information Forensics And Security, vol. 12, no. 3, pp. 676-688, 2017.

[4] Yujue Wang, Qianhong Wu, Bo Qin, Shaohua Tang, Willy Susilo, "*Online/offline provable data possession"*, IEEE transactions on Information Forensics and security, vol. 12, no. 5,pp. 1182-1194,2017.

[5] Hao Yan, Jiguo Li, Jinguang Han and Yichen Zhang C, "*A Novel Efficient Remote Data Possession Checking protocol in Cloud Storage"*, IEEE transactions on Information Forensics and security, vol. 12, no. 1,pp. 78-88, 2016.

[6] Sanjeet Kumar Nayak and Somanath Tripathy," *SEPDP: Secure and efficient privacy preserving*

provable data possession in cloud storage", IEEE transactions on Service computing, 2018.

[7] Debiao Hea, Neeraj Kumar ,Huaqun Wang, Lina Wang and Kim-Kwang Raymond Choo, "*Privacy Preserving certificateless provable data possession scheme for big data storage on cloud"*, Elsevier, Journal of Applied mathematics and computation, vol. 314, pp. 31-43, 2017

[8] Jian Shen, Jun Shen, Xiaofeng Chen, Xinyi Huang and Willy Susilo, "*An efficient public auditing protocol with novel dynamic structure for cloud data"*, IEEE transactions on Information Forensics and security, vol.12, no. 10,2016

[9] Huaqun Wang, Debiao He, Jia Yu and Zhiwei Wang, "*Incentive and unconditionally anonymous Identity-based public Provable Data Possession"*, IEEE Transactions on Service Computing, 2016

[10] Shiuan-Tzuo Shen, Hsiao-Ying Lin and Wen-Guey Tzeng, "*An effective integrity check scheme for secure erasure code-based storage systems"*, IEEE transactions on Reliability,vol. 64, no. 3,pp. 840-851, 2015

[11] Edaordo Gaetani, Leonardo Aniello, Roberto Baldoni, Federico Lombardi, Andrea Margheri, and Vladimiro Sassone, "*Blockchain-based Database to Ensure Data Integrity in Cloud Computing Environments*", https://securityintelligence.com/sabotage-the-latest-threat-to-the-financialbanking-industry/, 2016.