IOSR Journal of Engineering (IOSR JEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 78-82

Exploring Integrity Management and Verification Techniques for Cloud Data Storage

Ms. I. U. Wankhede*, Dr. S. S. Sherekar**, Dr. V. M. Thakare***
SantGadge Baba Amravati University, Amravati, M.S., India

Abstract: *Cloud Computing has emerged as a boosting technology for industries and communities all over the world. It has opened new vistas for researchers, developers, engineers and even common man. However, there are many issues which are yet to be addressed, most important of all, integrity and security management for cloud data storage. The Cloud Service Provider must perform a transparent role while keeping the user's data. It must provide some guarantee that the outsourced user data remains tamper proof. Various methods like Data Encryption, Auditing and Access control mechanisms have been proposed to maintain and check data integrity in cloud environment. Recently, Cloud Forensics has also gained importance as an integrity verification technique. The paper reviews few prominent data integrity management and verification techniques for cloud data storage. It explores the availability of integrity frameworks, protocols and tools applicable to the Cloud environment.*

Keywords: *Data Integrity, Cloud Data Storage, Auditing, Integrity Management*

I. Introduction

Cloud Computing has expanded manifold since its birth. It's being widely used to host applications, store and analyze huge datasets, providing scalable services and much more. In spite of this glorious progress, the dark shadows of threats and lack of transparency have always haunted the users. In context with the data storage services, the integrity and security of data has prime importance. Many researchers worldwide have suggested a wide array of approaches to ensure that only the authorized individual has access to data and the integrity and security of data is preserved along entire data cycle –the outsourcing, transit and storage. The Cloud Service Providers (CSPs) must be able to ensure transparency with regards to the integrity of user data**[1]**. In this paper we have discussed in brief various prominent integrity management and verification mechanisms for cloud data storage available today.

The paper is divided into sections. The Section 1 gives a brief introduction to the subject. The Section 2 highlights important works in the concerned field. Section 3 enlists the features of data integrity verification along with comparative study of various approaches. Section 4 discusses the pros and cons of the selected tools and frameworks compared in Section 3. The last section highlights the future prospects of research in cloud storage integrity maintenance and verification.

II. Background

Many mechanisms have been proposed till date to verify the integrity of the outsourced data stored on the cloud. However, serious shortcomings were detected in these schemes. Recently researchers have proposed novel approaches that overcome the year old shortcomings.

Wang et. al. [**2**] proposed a novel method for public auditing of data shard among the cloud users. This method completely preserves the identity of the users and their data during the process of verification. The scheme also supports batch auditing without requiring the user to download entire file. Zhang et. al. [**3**] proposed a computationally efficient public verification scheme that could handle dynamic data and perform batch verification.Jiang et. al. [**4**] proposed a remote data integrity auditing scheme which can resist collusion attack from CSP and users of shared data.Wen and Lei [**5**]put forward a novel idea of public verification through Attribute-based Encryption. This scheme protects the identity privacy of users by releasing fake secrets to the coercers. Linset. al. [**6**] argued that none of the earlier stated schemes is appropriate for third party auditing. The cloud environment is dynamic in nature and so is the data. They propose a conceptual continuous cloud auditing architecture.

Wang et. al. [**7**] have proposed an identity-based proxy-oriented data uploading and remote data integrity checking protocol for public cloud (ID-PUIC) which supports private, delegated as well as public verification of integrity. Shen et. al. [**8**] proposed an algebraic signature-based data integrity auditing scheme that ensures the cloud data integrity and confidentiality with batch auditing on dynamic data.

Yu et. al. [**9**] have proposed a remote data integrity checking mechanism based on bilinear pairing and Homomorphic encryption that guarantees zero knowledge reveal to the verifier. Mercier et. al. [**10**] introduced a

data storage architecture 'STEP-archives' based on data entanglement and erasure-correcting codes. In such storage a document can be deleted or modified only by bargaining the integrity of other documents in the storage. For any cloud platform to be secure and trusted, the individual layers of the platform must be secure.Suleet. al. [**11**] presented a multi-layer trust security model (MLTSM) based on unified cloud platform trust that employs a fuzzy logiccombination of on-demand states of several different security mechanisms, such as identification, direct and in-direct trust.

III. Integrity Management and Verification Techniques

The traditional mechanisms for remote integrity management and verification can be classified as follows-

3.1 Third Party Auditing

This is very popular approach**.** It needs a trusted unbiased Third Party Agent (TPA). This agent would certify the Cloud Service Provider (CSP) for integrity metrics whenever the user demands so. Traditional integrity verification demanded huge computational overheads [2]. It requires downloading entire file. To save the computational overheads, the file is divided into blocks and each block is signed individually. Thus, the user or a third party agent can check the integrity by making lesser computations. Such method is also termed as Public Auditing. The user itself or TPA issues a challenge to the CSP on behalf of the user. The CSP can prove the integrity of the data by successfully completing the challenge. However, there are few issues with his approach. There is always a risk that the TPA would be compromised. Many schemes fail to perform batch verification and usually face the challenges in sequential manner. Also, while verifying the integrity of data shared among the users, such schemes fail to preserve the identity privacy. In such case the users may collude with CSPs to leak data.Each verifier or the device implementing the verification scheme may always be computationallycapable. Another challenge is the time to time verification of dynamic data.

3.2 Encryption and Cryptanalysis

Digital Signatures and Hashes can be attested to check the integrity of the data. But in spite of being secure, these methods have been known to leak user data by snooping access patterns. Many Encryption Schemes facilitate only the user to validate the integrity. There shuld be some facility for delegated verification.

3.3 Proofs of Storage and Data Retrieval

Proofs of Retrievability (POR) are cryptographic tools that help a CSP to prove that a user can retrieve his file at any point of time. POR need to be frequently executed by the user to ensure that their files stored in the cloud can be fully retrieved. To conduct and verify POR, users need to have proper network access. As users use portable devices with limited computational capacity use low bandwidth network access, the implementation of POR by cloud users is usually difficult.

3.4 Digital forensics

The investigators face many challenges when they investigate crime occurred using cloud. Many researchers try to identify and solve these challenges by proposing new methods and techniques. The digital forensic techniques can be combined with data integrity to find useful evidences in case or crime.

3.5 Security Metrics and Service Level Agreements

The storage services that follow a security framework and satisfy the metrics can be trusted by the users. There is a need to develop such frameworks and design the proper metrics.

There are several challenges for we face while implementing an integrity verification management like -

- **Computational Overheads**: The stored data must be first setup for integrity management before you can perform integrity audit or verification of the data. On the other hand, verification also requires the execution of complex algorithm when an integrity challenge is received.
- **Disk I/O**: The cloud store large amount of data. The user would need to download an entire file before it can be verified for its integrity. This causes tremendous computational overheads. In other cases, at least a few blocks of data needs to be downloaded.
- **Time complexity:** Complex integrity verification needs to be executed in certain steps. This needs time to perform computations. Some algorithms need prior operations too.
- **Lack of bandwidth:**Heavy communication costs between auditor, CSP and user may pose hurdles as users have limited bandwidth.
- **Computational power of the devices:** The world around us isgetting portable day by day. The users may not always have a capable device with enough computational power to run the verification mechanisms.

IV. Comparative Analysis of novel integrity verification techniques

A comparative analysis of the methods that are put forth as solutions to the drawbacks in each of the above categories provides a clear picture.We have chosen the following properties of integrity for comparision-for comparision –

- **Data Privacy:** This property specifies whether the data of the user remains hidden from the auditor or verification algorithms.
- **Identity privacy:**This property specifies whether the identity of the user remains hidden from the auditor or other users when the data shared among different users is being checked for integrity.
- **Forgery:** This property tests if a false identity can be maliciously fabricated and used to deceive the verifier.
- **Multiple / Batch Processing:** This specifies whether the scheme is able to handle multiple challenges of integrity verification simultaneously.
- **Shared DataSupport:**This property specifies whether the scheme works on data shared by various users.
- **Dynamic Data support:** This specifies whether the scheme supports changes in data with time.

Oruta [2] supports public auditing along with both data and identity preservation. Such a scheme prevents the loss of identity of the user to the public verifier. This is a very important property for a verification scheme. The mechanism [3] using Indisguishability Obfuscation has an important property. The auditor's overhead in its batch verification scheme is independent of the number of verification tasks. Moreover, the proposed scheme also achieves data dynamic operations, which include insertion, deletion and updating. One scheme [4] supports the public data auditing along with outsourcing ciphertext database to remote cloud and support secure group users revocation to shared dynamic data. Most of the schemes use an external auditor or verifier to validate the integrity of data. However, [5] is an audit free service. It saves the effort of verification and data communication between the auditor and cloud server. Findings in [7] reveal that cloud data should be continuously audited to prove secure and reliable services.[7] is the first Identity based remote data integrity checking protocol which can achieve private remote data integrity checking, delegated remote data integrity checking and public remote data integrity checking based on the original client's authorization. In [8] the computation time of the dynamic operations increases linearly as the number of blocks grow. Also,a prior block first needs o be deleted if we want to update a data block. This adds to the computational overhead although still being practically tolerable. [9] can support data dynamics by using data index matrix operations. [10] Raises the problem of variable block size. Using fixed and large block size would cause wastage of storage while small block size increases number of I/O operations. This is true with every verification scheme.

Data Integrity	Performance characteristics					
Data Support Dyn-amic	Compu-tational overhead	Commu-nication Cost	Time compl-exity	Time compl-Time complex-ity	Disk I/O	
X	Low	Approx. 15 Kb	Avg. 1.5 sec	Avg. 1.5 sec	2.02GB	
✓	Light	-	Approx. 0.4 sec for single verification	Approx. 0.4 sec for single-verification	-	
✓	Heavy	Heavy	Approx 15 secs or a block	Approx 15 secs or a block	Heavy	
X	Heavy	Heavy	Poor	Poor	Heavy	
X	-	1504 + log2 *n* bits	Approx. 1.6 sec for 20 blocks	Approx. 1.6 sec for 20 blocks	-	
✓	Heavy	Average	Average	Average	Heavy	
X	Average	Average	30sec for 1Mb	30sec for 1Mb	-	
X	Heavy	Heavy	Heavy	Heavy	Heavy	

Table: Comparative Analysis of novel Data integrity verification techniques based on Data Integrity and Performance Characteristics

Sr. No.	Method	Technique(s) used	Forgi-ngtity	Iden-Priv-a-cy	Data Priv-acy	Multiple Auditing / Batch Processing	Shar-ed
1	**Wang et. al. [2]** (for 10 users with 100 elements per data block)	Ring Signatures and Homomorphic Authenticators	X	✓	✓	✓	✓
2	**Zhang et. al. [3]**	Bilinear Maps &Indisguishability Obfuscation	X	X	✓	✓	X
3	**Jiang et. al. [4]**	Bilinear Groups & Vector Commitment	Nil	X	✓	X	✓
4	**Wen and Lei [5]**	Deniable Attribute based Encryption	X	X	✓	X	✓
5	**Wang et. al. [7]**	Identity-based & Proxy Public Key cryptography	X	X	✓	X	✓
6	**Shen et. al. [8]**	Algebraic signatures	✓	X	✓	✓	X
7	**Yu et. al. [9]**	Identity based Signatures	X	X	✓	X	X
8	**Mercier et. al. [10]**	Data Entanglement and Erasure Codes	✓	X	✓	X	X

V. Conclusion

The main concern in adoption of cloud has always been the security and integrity issues. This research area shall remain active in future too.We have attempted to make a comparison of striking mechanisms proposed in the area of integrity management and remote verification for cloud data storage. The comparison relies on various characteristics of integrity verification mechanisms as well as their performance characteristics. We have discussed the issues related to the main stream of approaches followed for integrity verification.

We can deduce that very few mechanisms are better in terms of computational and time complexity. Nearly every mechanism needs heavy disk operations for setup and verification of challenges. None of the methods in our list supports data dynamics, shared data, identity privacy and unforgeability principles at the same time. The values of performance characteristics have not been specified numerically for some mechanisms. One of the most interesting challenges with respect to shared data is to design an integrity checking mechanism which would protect the identity privacy as well as support identity traceability in some situations. None of the studies schemes have any provision to prove that the copy of data being verified for integrity is the latest one. Every integrity verification scheme requires a third party auditor or verifier. There must be some facility which shall facilitate for integrity verification by user itself. This would add to the transparency in cloud service.

Cloud environment is naturally dynamic. However, very little research has been done on continuous auditing which suites the dynamic nature of the cloud. The integrity verification mechanisms need to improve and be more practical and economic.

VI. Future Scope

Newer security metrics and homogeneous Security Level Agreements would be helpful for the user to choose and trust the service. The mechanisms also must have a proper way to delete the old and discarded data.Future research also includesevaluation of overheadand characteristics of proposed mechanisms and their comparisons with other existingcloud trust models.

References

[1]. M. Ouedraogo, S. Mignon, H. Cholez, S. Furnell and E. Dubois, "Security transparency: the next frontier for security research in the cloud", Journal of Cloud Computing: Advances, Systems and Applications, Springer, June 2015

[2]. B. Wang, B. Li and H. Li, "Oruta Privacy-Preserving Public Auditing for Shared Data in the Cloud", IEEE Transactions on Cloud Computing, Vol. 2, No. 1, pp. 43-56, 2014.

[3]. Y. Zhang, C. Xu, X. Liang, H. Li, Y. Mu and X. Zhang, "Efficient Public Verification of Data Integrity for Cloud Storage Systems from Indistinguishability Obfuscation", IEEE Transactions on Information Forensics and Security, Vol. 12, No. 3, pp. 676-688, March 2017

[4]. T. Jiang, X. Chen, and J. Ma, "Public Integrity Auditing for Shared Dynamic Cloud Data with Group User Revocation", IEEE Transactions on Computers, Vol. 65, No. 8, pp. 2363 – 2373, August 2015

[5]. P. Wen Chi and C. Lei, "Audit-Free Cloud Storage via Deniable Attribute based encryption", IEEE Transactions on Cloud Computing, Vol. 6 , No. 2 , pp. 414 – 427, April 2015

[6]. S. Lins, S. Schneider and A. Sunyaev, "Trust is Good, Control is Better: Creating Secure Clouds by Continuous Auditing", IEEE Transactions on Cloud Computing, Vol. 6, No. 3, pp. 890 - 903, July-September 2016

[7]. H. Wang, D. He, and S. Tang, "Identity-Based Proxy-Oriented Data Uploading and Remote Data Integrity Checking in Public Cloud", IEEE Transactions on Information Forensics and Security, Vol. 11, No. 6, pp-1165-1176, June 2016

[8]. J. Shen, D. Liu, D. He, X. Huang and Y. Xiang, "Algebraic Signatures-based Data Integrity Auditing for Efficient Data Dynamics in Cloud Computing", IEEE Transactions on sustainable Computing, pp.1-1, December 2017

[9]. Y. Yu, M. Au, G. Ateniese, X. Huang, W. Susilo, Y. Dai, and G. 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, pp. 767 – 778, April 2017

[10]. H. Mercier, M. Augier and A. Lenstra, "STEP-archival: Storage Integrity and Tamper Resistance using Data Entanglement", IEEE Transactions on Information Theory, Vol. 64 , No. 6 , pp. 4233 – 4258, June 2018

[11]. M. Sule, M. Li1, G. Taylor and C. Onime "Fuzzy logic approach to modelling trust in cloud computing", IET Cyber-Physical Systems: Theory & Applications, Vol. 2 , No. 2 , pp. 84-89, July 2017