Volume 4, Issue 12, December 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Survey on Data Integrity Checking Techniques in Cloud Data Storage

Charmee V. Desai*, Prof. Gordhan B. Jethava
CSE Department,
Parul Institute of Engineering & Technology, India

Abstract-Cloud Data Storage is an attractive mean for storing and managing the outsourced data. As data are remotely stored, user is not aware of any security threat. Data modification can done by the untrusted server , unauthorized user or by some malicious activity. So user needs to be ensured that their data are intact. For this various integrity checking techniques for cloud data storage have been proposed. This paper presents a survey of different Integrity Checking Techniques for remotely stored data on cloud. Comparison of all techniques with their advantages and disadvantages are provided. Objective of this survey is to provide comparative analysis of existing integrity check techniques for cloud data storage.

Keywords- Cloud Data Storage, integrity checking.

I. INTRODUCTION

Cloud storage provides the powerful way of managing data. It allows to store and manage the data remotely. Users do not have to buy the expensive hardware and to have policies to regulate and manage the data. Apart from this users have to pay only for what they use. Because of the ubiquitous nature of the cloud, it allows to access the data from anywhere. These makes cloud storage very popular.

As cloud is distributed in nature, once user uploads the data user has no control over the data. Data are remotely stored. Users do not have physical access to the data. So user is not aware of any security threats at storage site. So to ensure the integrity of remotely stored data is major concern. various integrity check techniques have been proposed with their pros and cons. With integrity check techniques users can be ensured that their data are intact at storage site and not have been modified by an unauthorized entity.

In 2008 Amazon has faced the downtime after that, they were not able to recover the original data.[1] In 2006 Gmail has faced the mass deletion of email which resulted the loss of data. [2] Cloud Service providers like Amazon, places explicit statements like they are responsible for any kind of data loss or data damage to save their reputation. [3] Cloud service provider are not providing any integrity check technique explicitly. They are just providing the MD5 etag value which is updated whenever file is modified remotely. So it is necessary for user to check the integrity of their remotely stored data.

II. PRINCIPLES OF DATA INTEGRITY

In cloud everything is provided as service. Storage is also provided as service for e.g. Amazon Simple Storage Service. The Cloud storage architecture is as below.

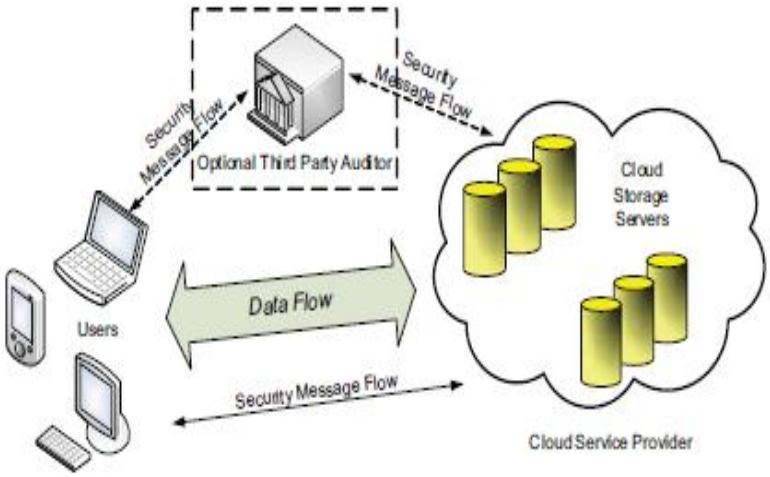

Fig. 1 Cloud Storage Architecture

As shown in fig. it consist of three entities: Client, Service Provider and Third party Editor.[4]

Client: They are the users who actually uses the services. Users has the data to store on cloud and rely on service provider for data management , data backup and other quality of service parameters.

Service Provider: They provide the services through protocol. They own the infrastructure and other elements with which they manage users , computation resources, data , security and manage all services.

The Third Party Auditor (TPA): It is an optional entity. It is responsible for querying the server on behalf of client. It has expertise and capabilities that users may not have. It is considered as trusted party. It queries the server upon the request of client and it mainly used for checking security parameters.

Integrity of the remotely stored data can be verified in two ways:

1)Private Auditability: In this client itself challenges the server for required parameter and verifies the integrity of data.

2)Public Auditability: In this anybody can challenge the server for the verification of data.

Below figure shows the core approach for Data Integrity Checking:

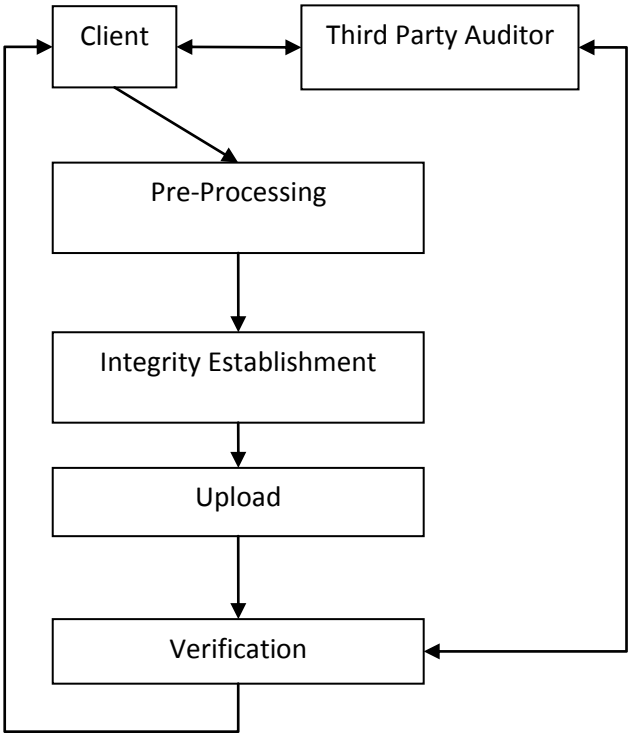

Fig. 2 Core Integrity Checking Approach

Step-1 Pre-Processing: In this step metadata is generated. For verification something has to be compared. Metadata is used for this purpose. There are various technique by which metadata can be generated. After that metadata is encrypted so that adversary is not able to read it in its original form.

Step-2 Integrity Establishment: In this step metadata is either stored on local machine or based on the method used for metadata generation , metadata is embedded along with the original file.

Step-3 Upload: In this step file is uploaded to the storage server.

Step-4 Verification: In this step ,client or TPA requests the required parameters and metadata for the integrity check of requested file from the server. Server responds with requested data to the client or the TPA. The verifier either compares with locally stored copy of metadata or recomputes the metadata and compares with the received one. If both matches then file is considered to be intact otherwise it is considered as tampered illegally.

III. METHODS FOR INTEGRITY CHECKING

[5] **Provable Data Possesion (PDP)**:In this client pre-computes tags for each block of a file. Then stores the file and its tags with a server. Later the client can verify that the server possesses the file by generating a random challenge against a randomly selected set of file blocks. Using the queried blocks and their corresponding tags, the server generates a proof of possession. The client is thus convinced of data possession, without actually having to retrieve file blocks.

[6]**Proof of Retrievability (PoR)**:In this scheme ,first file is divided into blocks and then encoded with error correcting codes. Then check blocks called sentinels are embedded for each block. Encryption is performed to make check blocks indistinguishable from other file blocks. The verifier challenges the prover by specifying the positions of a collection of sentinels. The prover returns the respective sentinels. If prover has modified or deleted a substantial portion of file, then with high probability , it will have also suppressed a number of sentinels. Using error correcting code file can be recovered.

[7] **MD5 based:**This method is based on MD5 function. In this method first file is read and compressed. This compressed content is input to the MD5 function which generates the message digest . This Message digest is encrypted and appended with the original file content within pre defined tag. For verification, file is read back. File is compressed

and MD5 is used to generate hash value. This hash value is encrypted and compared with appended one in original file. If both matches then file is intact otherwise it is tampered.

[8] **Encryption Algorithm:**In this method, proposed system is consist of three entities. A trusted third party called cloud broker, clients and cloud service provider. In this broker acts as interface between cloud data storage server and client. Request handling and integrity checking is performed by this entity. In this method, Client inputs a file. At broker side a partitioner partitions the file based on the storage available. Tag generator module generates the tag for each segments. Hash values for each segment is calculated and stored in database. Database manager stores all relevant and required values in database. For verification, verifier retrieves all encrypted segments from the storage and calculates the hash of all segments. These values are compared with respective segment's hash value stored in database. If hash of all segments matches than file is intact otherwise tampered.

[9] **Generate,Encrypt and Append Metadata:** This method is based on XOR operation. First the file is divided into blocks. Size of metadata is fixed. For each of the block, bits are selected from various positions. generation of these positions is done using a function g which takes the parameter as block no and 1 to size of metadata. Another function h is used to generate random byte sequence which is then XORed with values located at the positions generated by the function g. This XORed values are used as encrypted metadata and appended to the end of file. File is then uploaded.

For Verification, Let the client wants to verify the integrity of ith block. It sends the block no and the bit no generated by function g. Client also specifies the position at which the metadata for the block is appended. Server responds with specified metadata and the corresponding bit. At client side decryption is performed to generate the data which is then compared with bit sent by the server. Any mismatch shows the loss of integrity.

[10] **RSA based:**In this, method specified is based on the RSA algorithm which makes use of large prime numbers. It has four phases KeyGen , SigGen, GenProof, VerifyProof

KeyGen: Public key and private keys are generated. File is encrypted using the key. Then file is divided into blocks. For each of the block authenticators are generated.

SigGen: Generates verification metadata. This function generates the authenticator for each of the block of file.

GenProof: Run by cloud server to generate a proof of data storage correctness

VerifyProof: Run by the TPA to verify the proof from the cloud server

IV. COMPARISON OF EXISTING METHODS

TABLE I

Reference Paper	Integrity Check Method	Advantage	Disadvantage
[5]	PDP	Data need not to be downloaded for verification	Data recovery is not supported
[6]	PoR	Data recovery is possible,	Can not be used in original form, pre processing is required for encoding
[7]	MD5 based	Low computation overhead, Suitable for small size	No public auditability, Data download needed for verification ,No support for dynamic operation
[8]	Encryption Algorithm	Supports Confidentiality, low computation and storage overhead at client side.	No support for dynamic operation, No public auditability, Uses only 4 bit key
[9]	Generate, Encrypt and append metadata	Low computation overhead, Low storage overhead	No support for dynamic operation, No public auditability
[10]	RSA based	Supports public auditability	Due to exponential operation computation overhead.

V. CONCLUSION

In this paper various methods for integrity checking for remotely stored data on cloud are discussed. It is observed that various methods for integrity checking are available . So depending upon the file size and requirement methods can be used. At last the comparative table with their advantage and drawbacks of all the methods are provided.

ACKNOWLEDGMENT

I would like to express my sincere thanks to my guide Prof. G.B. Jethava (Head, IT Department), for his great efforts and encouraging for this work. I am very much thankful for his continuous guidance & support. I would like to thanks all my friends and family members for their support.

REFERENCES

[1] Amazon.com, "Amazon s3 Availability Event: July 20, 2008," July 2008. http://status.aws.amazon.com/s3-20080720.html

[2] M. Arrington, "Gmail Disaster: Reports of Mass Email Deletions," Dec. 2006. http://www.techcrunch.com/2006/12/28/gmail-disasterreports-of-mass-email-deletions/

[3] http://aws.amazon.com/agreement/

[4] T.S.Khatri,Prof. G.B.Jethava "Survey on Data Integrity Approaches used in the Cloud Computing " *International Journal Of Engineering Research & Technology* vol.1,pp. 1-6 November 2012

[5] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," *Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07),* pp. 598-609, 2007.

[6] A. Juels and B.S. Kaliski Jr., "Pors: Proofs of Retrievability for Large Files," *Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07),* pp. 584-597, 2007.

[7] Sanchika Gupta,Anjani Srdan, Padam kumar,Ajit Abraham,"A Secure and Lightweight Approach for Critical Data Security in Cloud" in *Proc.CASoN-IEEE* Jan 2012

[8] Pa.Varalakshmi#1, Hamsavardhini Deventhiran#2, "Integrity Checking for Cloud Environment Using Encryption Algorithm," in *Proc. ICRTIT-IEEE* 2012

[9] Sravan Kumar R, Ashutosh Saxena, "Data Integrity Proofs in Cloud Storage," in *Proc. COMSNETS-IEEE* Jan,2011

[10] Wenjun Luo, Guojing Bai,"Ensuring The Data Integrity in Cloud Data Storage " in *Proc. CCIS-IEEE* Jan 2011

