A Comparative Study on Cloud based Data Integrity Verification Schemes

Sonali Ghule
Walchand Institute of Technology
Solapur

Pratibha Yalagi
Walchand Institute of Technology
Solapur

ABSTRACT
Cloud computing is the delivery of cloud services over the Internet. Cloud services becoming popular because people are able to access their email, social networking site from anywhere in the world, at any time, at minimal charge or no charge. Cloud storage allows users to store their data remotely and use the on-demand high quality cloud applications without load of local hardware and software management. It moves the application software, databases and the important data to the centralized huge data centers, where the management of the data and services may not be fully secure. When user store their data on the cloud, there may be a risk of losing the data, or sometimes data may be modified or updated. It may not be fully secure because the client does not have copies of all stored data. To protect outsourced data against corruption enabling data integrity protection, fault tolerance and efficient recovery for cloud storage is required. This paper delivers a survey about, different data integrity techniques and their limitation. The data integrity techniques for privacy preservation are POR (Proof of Retrivability), PDP (Provable Data Possession), HAIL (High Availability and Integrity Layer for Cloud Storage), erasure codes etc.

Keywords
Data integrity, Proof of Retrievability, Provable Data Possession, Replication Based system

1. INTRODUCTION
Cloud Computing refers to manipulating, configuring, and accessing the applications online. Cloud is nothing but large groups of remote servers. These remote servers are networked to allow online access for data storage and services which is centralized. One major use of cloud storage is long-term archival, where data is stored that is written once and rarely read; it remains necessary to ensure its integrity for disaster recovery. The data integrity proofs the validity, consistency and regularity of the data. Integrity is the guarantee by which the data is protected from accidental modification. Therefore cloud storage is becoming popular for the outsourcing of day to day management of data. So checking the data integrity of the data in the cloud is also very important to remove all possibilities of data corruption and data crash.

There are basically two different data integrity proving schemes. POR (Proof of Retrievability) [1] and PDP Proof of Data Possession [2].These two schemes are used in the single server setting. But putting all the data in single server leads to the single point of failure problem [3] and vender-lock-ins [4].To avoid this one solution is to stripe the data across multiple servers. If one of the servers gets failed then to repair a failed server, they can first read the data from surviving servers reconstruct the corrupted data of the failed server and finally write the reconstructed data to the new server. For this purpose, MR-PDP [5] and HAIL [6] integrity checking schemes are used.

MR-PDP [5] and HAIL [6] are integrity checking scheme which is used in a multi server setting using replication and erasure coding [7], respectively. Erasure coding has a lower storage overhead than replication under same fault tolerance level.

2. DATA INTEGRITY SCHEME FOR SINGLE SERVER
There are two different types of data integrity scheme for single server. These are provable data possession i.e. PDP and proof of retrievability i.e. POR.

2.1 Provable Data Possession (PDP)
The file consists of collection of n blocks. A file is retained by a remote cloud server is checked by this PDP scheme. To generate some metadata the data owner processes the information file and stores it locally. After sending files to the server, the owner deletes the original copy of the file. The owner verifies the possession of a file. The client uses this technique to check the integrity of the data. So this technique ensures server security to the client.

Compare the data is the main idea behind this scheme. With the file F and having key K (i.e. (K, F)), the client will compute the hash value. After computing hash value, it will send the file F to the server. Clients are having a different collection of keys and hash values so that it can check multiple checks on the file F. The client sends the key K to the server whenever it wants to check the file, which is then asked to recompute the hash value using F and K. With hash value for comparison server will reply back to the client.

Although this method gives the proof that the server is having the original file F, this method has high overhead because every time hashing process is run over the entire file. Hence, it is having very high computational cost.

2.2 Proof of Retrievability (POR)
Proof of Retrievability scheme proposed by Juels and Kaliski [1]. To verify the data stored by user on remote storage in the cloud is not modified by the cloud Proof of retrievability scheme is used. This scheme verifies the integrity of large files via various cryptographic primitives. It's a scheme which does not involve the encryption of the whole data. It reduces the computational overhead on the client side by encrypting only a few bits of data per data block.

It does not store any data on client side so the client storage overhead is also minimized. Hence this scheme suits well for thin clients [10]. This scheme reduces the computational as well as storage overhead of the client and the server. It reduces the network bandwidth consumption by minimizing the size of the proof of data integrity.

There is another POR scheme for the huge size of files named as sentinels. The main role of sentinels is cloud needs to

access only a small portion of the file (F) instead of a whole file. In this scheme the file is divided into a number of blocks. In this scheme the client stores only a single cryptographic key. This key is selected irrespective of the size and number of the files. This POR scheme encrypts F. Later on randomly embeds a set of randomly-valued check blocks called sentinels into an encrypted file [1].The use of encryption here makes the sentinels indistinguishable from other file blocks. The client challenges the server by specifying the positions of a collection of sentinels and asking the server to return the associated sentinel values.

The above both schemes are single server storage schemes. The problem arises in this method are vendor-lock-ins [4] and single-point-failure [3]. These problems can be overcome by striping data across multiple servers. To check data integrity for multi-server settings MR-PDP [9] and HAIL [6] schemes are used which are based on replication and erasure coding techniques respectively.

3. DATA INTEGRITY SCHEME FOR MULTISERVER SETTINGS

The files are striped and redundantly stored across multiserver. For this there is a need to explore integrity verification schemes suitable for such multiserver settings with different redundancy schemes, such as replication, erasure codes and regenerating codes

3.1 Replication Based System

Redundancy is used for establishing reliability. The simplest form of redundancy is nothing but the replication. This is used in many storage systems in which t identical copies of each data object are kept at each instant by system members. If one node fails, then to repair failure node only single copy of file is required. i.e. if any node fails, then simply copy the replica of that file from surviving node and store it on the new node. For any replication based system the storage cost is very high.

3.1.1 MR-PDP Scheme

For the assurance and availability of data file on unsecure storage systems like clouds, many storage systems rely on replication.MR-PDP is the scheme used replication to store the data in the cloud. MR-PDP is a provably-secure scheme that stores t replicas of a file in the storage system. That allows the client to verify that each unique replica can be produced. And storage system uses t times the storage required to store a single replica. In this first create different replicas or copies of the data file by first encrypting the file. After that masking the encrypted version with some randomness which is generated from a Pseudo-Random Function (PRF) is being performed in MR-PDP [5]. MR-PDP is the extension of previous work on data possession proofs for a single copy of a file on a client/server storage system.MR-PDP scheme is computationally more efficient than single-replica PDP scheme. One more advantage of MR-PDP is that it can generate replicas on demand. But the disadvantage is that there is little expense when some of the existing replicas fail.

3.2 Erasure Coding Based (Reed Solomn Code) system

Erasure coding creates a mathematical function that describes a set of number so that they can be checked for accuracy and recovered if one is lost. This is the main idea behind erasure coding methods. The erasure codes are implemented most often using Reed-Solomon codes. Erasure coding offers better storage efficiency than Replication Based System. Suppose, file of size M can be divided into k pieces, i.e. into fragments, each of size M/k, encode them into n fragments of the equal size using an (n, k) maximum distance separable (MDS) code, and store them at n nodes. The original file can be recovered from any set of k coded fragments. In case of storage efficiency, it is storage cost effective, because k pieces each of size M/k provide the less data for recovering the lost file.

3.3 HAIL Scheme

HAIL (High Availability and Integrity Layer) is a data integrity scheme that allows a set of servers to prove to a client that a stored file is not modified and retrievable. HAIL is different from the other techniques those have been discussed so far. HAIL allows the client to store their data on multiple servers so there is a redundancy of the data. And at the client side only small amount of data is stored in local machine. This technique support for static data and does not applicable for dynamic data blocks. It is possible to check data integrity in the distributed storage via data redundancy. Here proof is generated that is independent of the data size and it is compact in size. HAIL uses the pseudorandom function, message authentication codes (MACs), and universal hash function for the integrity process. [6]

3.4 Regenerating Coding Based system

In erasure coded system repair from single node failure is done by reconstructing the whole data object to generate just one data block. But this is an inefficient way of regeneration because it uses more network bandwidth. Regenerating code has been proposed to minimize this repair traffic. That means they minimize the amount of data being read from remaining healthy servers. They achieve this by reading a set of chunks smaller than the original file from other surviving servers and reconstructing only the corrupted or lost data chunks. Regenerating coding based system is more efficient than an erasure coded system.

4. COMPARATIVE STUDY RESULTS

Table 1. Comparative Study Table

Data Integrity verification Scheme	Methodology/Algorithm used	Advantages	Limitations
Provable Data Possession	Key Generation Algorithm	This scheme gives strong data integrity verification. PDP reduces Block accesses. It also reduces server computation overhead and network traffic	Does not use error-correcting codes (EEC) to address concerns of corruption. Does not support privacy preservation. Applicable for static data only.

PDP Scheme based on MAC	Message Authentication Code(MAC)	This is simple & Secure scheme. Gives strong data integrity verification.	Support limited number of verifications with limited number of secret keys. The client has to retrieve the entire file F from the server to compute new MACs, which is not possible for large file.
POR for large files	Sentinel-based scheme	This scheme ensures both possession and retrievability of files on any storage.	Newly inserted sentinels and error correcting codes put computational overhead [1] Preprocessing/encoding of file required prior to storage with the server. Works only with static data only and no dynamic support.
High Availability Integrity Layer (HAIL)	MAC(Message Authentication Code), Pseudorandom function, Hash Function	Low storage overhead High availability and tolerance to adversarial failures[6] Enables client-side integrity checks	This technique is only applicable for the static data and not for dynamic update of block
MR-PDP	Signature aggregation	Each unique replica can be generated at the time of the challenge, it can generate further replicas on demand [5]	Unable to address how the authorized users can access the file copies from the cloud servers [11]. Computation overhead on both the client and server side Storage overhead

The above comparative study table "Table 1" gives consolidated schemes of various availability and integrity verification schemes along with their methodology or algorithm used in each scheme. The Comparative study Table of various schemes and the algorithm or implementation methodology that have used for each scheme is tabulated in Table 1. The advantages and limitations of each data integrity scheme are also specified in the table. The comparative study table provides the importance of data integrity verification that the client need to be done before storing their data to the third party server.

There are two fundamental approaches to client verification of data availability and data integrity. These are PDP and POR. In PDP based on a key generation algorithm, a client that has stored data on an untrusted server can verify that the server store the original data without retrieving it. While PDP based on the MAC is simpler than PDP based on a key generation algorithm. In PDP based on a key generation algorithm, the client generates pairs of matching keys public & secret key. While in PDP based on MAC, client generates message authentication code along with a set of secret keys. In POR based on encryption, few data bits per data block are encrypted instead of a whole file. This scheme is well suited for thin clients, because the data is not stored at client side. While in POR based on large file error correcting codes are employed to protect against corruption. POR scheme requires preprocessing steps that the clients should do before sending their file to cloud storage provider. But this is not suitable for updating the data efficiently. There is an improved version of this scheme has been proposed called compact POR. This technique uses homomorphic property to aggregate a proof into authenticator value but use for static data only. Several POR schemes and models have been proposed by using RAID techniques.The PDP (Provable Data Possession) and POR (proof of retrievability) schemes are useful for single server settings. For multiserver settings MR-PDP and HAIL schemes are used. HAIL is the extension of the basic single server design of PORs and does not suitable for the thin client. HAIL allows to store the data on many servers hence there is redundancy of the data. In HAIL scheme at the client side only small amount of data is stored.MR-PDP is the extension of the simple PDP scheme based on replication. In MR-PDP, there is computation overhead because of replicas, but this is not the case in PDP. for thin clients, because the data is not stored at client side. While in POR based on large file error correcting codes are employed to protect against corruption. POR scheme requires preprocessing steps that the clients should do before sending their file to cloud storage provider. But this is not suitable for updating the data efficiently. There is an improved version of this scheme has been proposed called compact POR. This technique uses homomorphic property to aggregate a proof into authenticator value but use for static data only. Several POR schemes and models have been proposed by using RAID techniques.

5. CONCLUSION

To improve productivity and reduces costs cloud computing offers great potential. Though cloud computing offers many

advantages, it also imposes security challenges which relate to cloud storage. The main purpose of using the cloud is to store the data. After storing or uploading the data to the server, the client will lose the control of the data, so data integrity is the main issue of the client side. Many efforts had been conducted to ensure the integrity of data in cloud storage. To outsource the data in the cloud, the security will be provided by the encrypted format. Only the authorized person can access the outsourced data. Nowadays, many techniques available out of which this paper analyzed Provable Data Possession (PDP) and Proof of retrievability (POR), MR-PDP (Multiple-Replica Provable Data Possession), HAIL (High-Availability and Integrity Layer). In this paper different existing data integrity techniques and their advantages and limitations are explained. The analytical study briefly compares all these techniques. These techniques can be manipulated to reduce the storage overhead of the client and to minimize the computational overhead of the remote storage server. New techniques can be designed to minimize the size of the data integrity proof, so as to minimize the network bandwidth consumption. The insider/outsider attacker or intruder can corrupt the images and videos. So as a future work can focus on providing integrity protection to images and videos.

6. REFERENCES

[1] A. Jules and B. Kaliski Jr., "PORs: Proofs of Retrievability for Large Files," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), 2007.

[2] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, L. Kissner, Z. Peterson, and D.Song, "Remote Data Checking Using Provable Data Possession," ACM Trans. Information and System Security, vol. 14, article 12, May 2011.

[3] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp 50-58, 2010.

[4] H. Abu-Libdeh, L. Princehouse, and H. Weatherspoon, "RACS: A Case for Cloud Storage Diversity," Proc. First ACM Symp. Cloud Computing (SoCC '10), 2010.

[5] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "MR-PDP: Multiple-Replica Provable Data Possession," Proc. IEEE 28th Int'l Conf. Distributed Computing Systems (ICDCS '08), 2008.

[6] K. Bowers, A. Juels, and A. Oprea, "HAIL: A High-Availability and Integrity Layer for Cloud Storage," Proc. 16th ACM Conf. Computer and Comm. Security (CCS '09), 2009.

[7] I. Reed and G. Solomon, "Polynomial Codes over Certain Finite Fields," J. Soc. Industrial and Applied Math., vol. 8, no. 2, pp. 300-304, 1960.

[8] Y. Hu, H. Chen, P. Lee, and Y. Tang, "NCCloud: Applying Network Coding for the Storage Repair in a Cloud-of-Clouds," Proc. 10th USENIX Conf. File and Storage Technologies (FAST'12),2012.

[9] R. Li, J. Lin, and P. Lee. CORE: Augmenting Regenerating-Coding-Based Recovery for Single and Concurrent Failures in Distributed Storage Systems. arXiv, preprint arXiv:1302.3344, 2013.

[10] Sravan Kumar R, Ashutosh Saxena, Data Integrity Proofs in Cloud Storage, 978-1-4244-ss8953-4/11/@ 2011 IEEE.

[11] Ayad F. Barsoum and M.Anwar Hasan, Provable Possession and Replication of Data over Cloud Servers