ISSN No. 0976-5697

Volume 5, No. 5, May-June 2014

International Journal of Advanced Research in Computer Science

CASE STUDY AND REPORT

Available Online at www.ijarcs.info

# On Data Integrity in Cloud Storage Services

Anjali A
M.Tech, CSE Dept.
Viswajyothi College Of Engineering And Technology
Vazhakulam, Ernakulam,India

Joe Mathew Jacob
Asst. Prof. CSE Dept.
Viswajyothi College Of Engineering And Technology
Vazhakulam, Ernakulam,India

***Abstract:*** The introduction of cloud computing has triggered a change in computer community. The varying requirements of users are faultlessly met by the cloud. The cloud is used for a variety of purposes including storage services. Cloud storage services allow users to store their data and thus reduce space consumption. But security and privacy of data stored in cloud is a major issue faced by the user's and service providers alike. This paper deals with the threats faced by the storage service providers and users in the area of data storage in cloud. The paper then proceeds to outline the need for auditing protocols and the challenges faced by them when it comes to the context of cloud. The paper concludes with an introduction into some of the cloud auditing protocols

***Keywords:*** Auditing; Cloud Computing; Data Storage; Security; Integrity; Bilinear pairing.

## I. INTRODUCTION

Cloud computing is an idea introduced to refer to 'distributed computing' over Internet (or any real time network). The expression cloud is commonly used in science to describe a large agglomeration of objects that visually appear from a distance as a cloud. Cloud computing is also the agglomeration of a variety of technologies and services that can be made available on demand. NIST [1] defined Cloud Computing as "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction". Increase in usage of Cloud and the various services provided by cloud has lead to some issues of concern that include reliability, availability of services and data, security, complexity, costs, regulations and legal issues, performance, migration, reversion, the lack of standards, limited customization and issues of privacy.

With the evolution of digitized data, our society has become dependent on services to extract valuable information and enhance decision making by individuals, businesses, and government in all aspects of life. Therefore, emerging cloud-based infrastructures for storage have been widely thought of as the next generation solution for the reliance on data increases. In nutshell storage services refer to the variety of cloud server that provide their users with services that allows them to store valuable data such as email, family photos and videos, and disk backups. These services had an immediate and notable effect and users began to store large volumes of data in cloud, or in cloud service providers. These CSPs (Cloud Service Providers) were usually companies with big-end servers and systems like Amazon, Google, Yahoo etc. Originally the CSPs started renting out their storage space to customers. But even such big-end servers have limited (though very large) amount of memory. As the number of customers increase this memory is very likely to get depleted. With this inference came thie issue of data integrity in the cloud.

A CSP that provides storage services may make a guarantee about the level of integrity it can provide. Failing behind in such standards may affect the reputation of the CSP and thus it is assumed throughout the paper that a CSP may resort to any means (legal or not) to ensure its users that their data is securely stored. But in reality that data may have been modified or completely or partially deleted. Such concerns lead to the need for privacy preserving and integrity checking protocols and mechanisms in Cloud.
The rest of the paper is organized as follows. In section II the threats faced by the data stored in cloud is analyzed. Section III gives an overview on data integrity and the challenges faced in the area. Section IV introduces the auditing protocols. Section V gives a brief introduction into various auditing protocols that are in existence today.

## II. THREATS TO DATA IN CLOUD

There are a variety of cloud storage systems. Some have a very specific focus, such as storing Web e-mail messages or digital pictures. Other such systems are available to store all forms of digital data. Size of a cloud storage system can be small or very large. The facilities that house cloud storage systems are called data centers. At its most basic level, a cloud storage system needs just one data server connected to the Internet. A client which can be a computer user subscribing to a cloud storage service sends copies of files over the Internet to the data server, which then records the information. When the client wishes to retrieve the information, he or she accesses the data server through a Web-based interface. The server then either sends the files back to the client or allows the client to access and manipulate the files on the server itself.

Data storage in cloud is an important service in the present computing world. Data storage services [2] allow any user to store any amount of data in cloud, or more specifically with a particular CSP. Storage services based on such Service Providers allows customers to move their data from their local machines. This will be advantageous to the customers as they can avoid the cost of building and maintaining a private storage infrastructure opting instead to pay a service provider as a function of its needs. For most of

the customers this provided several benefits of which the most important ones are availability and reliability at a low cost. Availability ensures that the data can be accessed from anywhere at any time. Reliability is the avoidance of the need to take and maintain backups. These two advantages make such services an attractive option. Cloud storage can also provide the benefits of rapid deployment, strong protection for data backup, archival and disaster recovery purposes and lower overall storage costs as a result of not having to purchase, manage and maintain expensive hardware.

Such benefits come with a few disadvantages. The most serious among them is security. Implementing a cloud storage strategy means placing critical data in the hands of a third party, so ensuring that the data remains secure both at rest (data residing on storage media) as well as when in transit is of paramount importance. A straightforward way is that data is kept encrypted at all times, with clearly defined roles when it comes to who will be managing the encryption keys. In most cases, the only way to truly ensure confidentiality of encrypted data that resides on a cloud provider's storage servers is for the client to own and manage the data encryption keys.

Data resting in the cloud needs to be accessible only by those authorized to do so, making it critical to both restrict and monitor who will be accessing the company's data through the cloud. In order to ensure the integrity of user authentication, companies need to be able to view data access logs and audit trails to verify that only authorized users are accessing the data. These access logs and audit trails additionally need to be secured and maintained for as long as the company needs or legal purposes require. As with all cloud computing security challenges, it's the responsibility of the customer to ensure that the cloud provider has taken all necessary security measures to protect the customer's data and the access to that data.

Data integrity implies that data should always be available as a whole – that is without any modification or whole or partial deletion. Ensuring data integrity is in itself a major issue in any data storage systems. When it comes to cloud, the issue is more profound. Data loss can occur with a higher probability here. This can be because of some problem in the infrastructure. Infrastructure problems can occur in any environment and is unavoidable to some extent. Another problem in cloud is integrity of data. Data is assumed to be safely stored in cloud by the users. But in such cases it is risky to trust a service provider. This is because any server or service provider functions with the main aim of maintaining its credibility.

## III. DATA INTEGRITY - CHALLENGES

Data integrity refers to the validity of the data at any point of time. In general, it refers to the idea of ensuring that the data is correctly stored in a server. It requires guarantee about accuracy and consistency of data. It requires the need to ensure that no data stored by any owner should be altered in any way, including manipulation or deletion (partially or completely).

One of the problems with any large data storage systems in general and Cloud data storage systems in particular is the increase in the volume of data. Substantial amount of data could get stored at the Server over time. In such cases, when the amount of data to be handled, or in the simplest case, stored, at the Server becomes too huge, the chances of data loss increases, knowingly or unknowingly.

Data loss can be either due to the fact that the Server cannot handle such large volume of data. Or it can also be due to the fact that the server knowingly drops the data, usually in order to

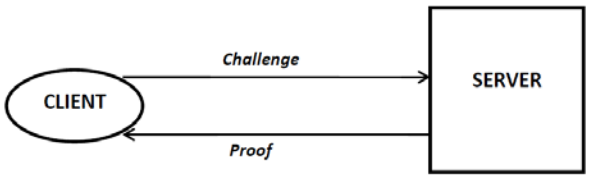


Figure. 1 Two party auditing protocol

Create more space. In such cases, Server always selects those data for deletion which have not been accessed for a considerable amount of time. This arises from the assumption that if a User has not requested for a particular data for a considerable amount of time, then the chances of that User requesting that particular data in future is negligible.

There can also be another problem mainly in Cloud data storages. In fact this problem is common in any scenario where one cannot put their complete trust in a Server. This means that, if the Server in question is not trustworthy, there is a possibility that the server can manipulate the data stored by the User. In such cases there is no problem of data loss since deletion of any stored data can be easily detected. Here emphasis is given to data manipulation. This includes appending, modifying, altering, or deleting some parts of the original version of the data stored by a User. Here the data entrusted to a Server is being misused.

Some other problems relating to data integrity may include accessing of information by unauthorized users which may lead to misuse of data. It may not always be a Server that manipulates data. It is also possible for an attacker to behave like a naïve User and make changes to the data. In such cases without proper authorization data can be misused.

## IV. DATA INTEGRITY - CHALLENGES

A major breakthrough occurred with the introduction of the idea of auditing protocols. In its simplest sense, an auditor or an auditing protocol does some operations to check whether the data stored at any Server is safe, in the sense that any type of manipulation, alteration or deletion can be identified by it.

The simplest type of auditing protocol that came into existence is the Two Party Authentication Protocol shown diagrammatically in figure 1. Here a Client is any authorized user of the system. A Client wishing to check the integrity of data stored by it in the Server will first issue a ‘challenge’ to the Server. This ‘challenge’ message will contain information about the file to be checked for. It can be, in the most general case, a File ID to denote the file. On receiving such a ‘challenge’ from the Client, the Server calculates a Proof which it sends back to the Client. From this Proof, the Client can verify whether the data is safe or not.

Though the Two Party Auditing Protocol became popular, it had several shortcomings. The computational load will increase both at the Client and the Server. Also, when it comes to the concept of large data storage systems, especially Cloud Storage Systems, one cannot guarantee that

all Users are naïve. There can be a case where a Client can become malicious and blame a trustworthy Server. Two Party Auditing Protocols has the general assumption that Clients are naïve. This can be compromised.

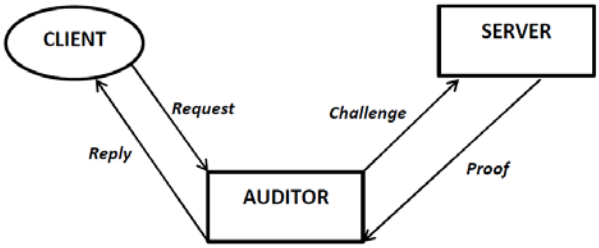


Figure 2. Third Party Auditor

These shortcomings led to the development of another idea that resulted in the general class of protocols called Third Party Auditing Protocols as shown in figure 2. These groups of protocols make use of another third party, apart from the Server or the Client, to do the auditing process. These chosen Third Parties' has the trust of both Client and the Server, and remains impartial. In such a Third Party Auditing Protocol, the Client may send a request for auditing to the Auditor, or the Auditor may initiate the Auditing by itself. In the latter case, Auditing process is usually done periodically to ensure safety of data stored by a Client. In either case the Auditor is the one that issues a Challenge to the Server, which will calculate a Proof and sends it back to the Auditor. The Auditor will then do the Verification to ensure that the data is correctly stored, and only the final response is send back to the Client. In this way safety and trust, both can be ensured in any Storage Systems.

When it comes to the case of a Cloud Storage System, it becomes more difficult. This is because the Storage System in consideration is so huge and vast that normal method for ensuring safety does not work well. So in such cases it is required to find solutions that will work despite the obvious challenges posed in a Cloud Environment.

Over the past few years, several auditing protocols where introduced and developed, keeping in mind the problems in the Cloud Environment. Most of these protocols relay upon mathematical properties which ensures safety of data. In the next session some of the popular auditing protocols in Cloud are discussed and compared.

## V. AUDITING PROTOCOLS IN CLOUD

Auditing in Cloud is a popular issue in discussion. Auditing is a complex process in the context of Cloud. This is so because one has to deal with a storage system that is large and complex in its own. The Servers usually provided for Cloud Storage do not encourage large scale computations. Thus the protocols must ensure that the computational complexity is kept to a bare minimum at the Server side.

'Toward Publicly Auditable Secure Cloud Data Storage Services' [4] deals with the basic requirements and also the challenges faced by auditing protocols in Cloud. According to [4], the design should be cryptographically strong, and more important, be systematic and practical . They further outline a set of suggested desirable properties. The most important one among them, as mentioned before is minimizing auditing overhead. The overhead imposed on the cloud server by the auditing process must not outweigh its benefits. Such overhead may include both the I/O cost for data access and the bandwidth cost for data transfer

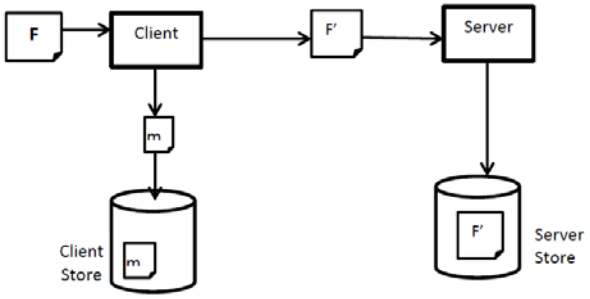


Figure 3. Preprocessing Step

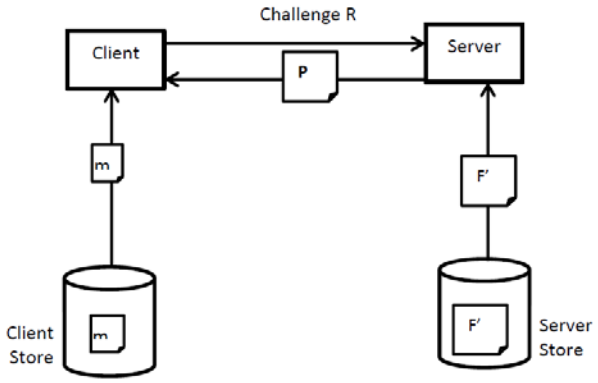


Figure 4. Verfication Step

Any extra online burden on a Client should also be as low as possible. Another issue is about protecting data privacy. The implementation of a public auditing protocol should not violate the owner's data privacy. If the Client has to share the contents of its critical data with the auditor, the term 'secure storage' will lose its meaning. As a cloud storage service is not just a data warehouse, owners are subject to dynamically updating their data via various application purposes. The design of auditing protocol should incorporate this important feature of data dynamics in Cloud Computing. The prevalence of large-scale cloud storage service further demands auditing efficiency. The paper then follows to outline the idea of using homomorphic tags, or authenticators to ensure that Client does not have to reveal the entire data content to the Auditor. Homomorphic authenticators are metadata that are not forgeable generated from individual data blocks, which can be securely aggregated in such a way to assure a verifier that a linear combination of data blocks is correctly computed by verifying only the aggregated authenticator.

'Provable Data Possession at Untrusted Stores' [5] is one of the earliest techniques that were developed for integrity checking in Cloud using homomorphic tags. Provable Data Possession, referred to as (PDP), becomes employed through the process of checking the data integrity with cloud storage. It allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data

checking supports large data sets in widely-distributed storage systems. Because of the homomorphic property, tags computed for multiple file blocks can be combined into a single value. The client pre-computes tags for each block of a file and then stores the file and its tags with a server. This is diagrammatically shown in figure 3. Here an input File F, to a Client is made to undergo some pre-processing steps and gets converted into the pre-processed file F'. Also a metadata that is unique to every file is stored at the Client.

At a later time, 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. This is shown in Figure 4, where R is the challenge issued by the Client and P is the proof calculated and returned back by the Server. This proof P is verified by the Client. The client is thus convinced of data possession, without actually having to retrieve file blocks. The advantage of such a scheme is that both plain and encrypted data can be checked using the idea of homomorphic tags. However one major disadvantage of this scheme is that it works only for static data. This is not advisable in a Cloud environment since dynamic operations are bound to happen at any point of time.

A variant of the original PDP scheme [5] was introduced in [6]. 'Scalable and Efficient Provable Data Possession' [6] Scalable PDP is an improved version of the original PDP. The differences in the two can be summarized as follows: One is that scalable PDP adopts symmetric key encryption instead of public-key to reduce computation overhead, but scalable PDP does not support public verification due to symmetric encryption. Another is that scalable PDP has added dynamic operations on remote data. One limitation of scalable PDP is that all challenges and answers are pre-computed, and the number of updates is limited.

The scheme is based on a symmetric-key cryptography method. Before outsourcing, the Owner pre-computes some short verification tokens, each token covering some set of data blocks. The actual data is then handed over to the Server. Subsequently, when Owner wants to obtain a proof of data possession, it challenges Server with a set of random-looking block indices. In turn, Server must compute a short integrity check over the specified blocks (corresponding to the indices) and return it to Owner. For the proof to hold, the returned integrity check must match the corresponding value precomputed by Owner. The scheme is very efficient in terms of computation and bandwidth. However, the main disadvantage of this approach is that dynamic operations require Server to send all unused tokens back to Owner resulting in the bandwidth overhead. But this overhead is unavoidable in order to ensure security.

After the introduction of the original PDP scheme [5], many variants were introduced. Each of them tends to overcome some shortcomings of the original scheme. Another variant introduced was the 'Dynamic Provable Data Possession' [7].

The paper provides a framework for dynamic provable data possession (DPDP), which extends the PDP model to support provable updates on the stored data. An update is defined as either insertion of a new block anywhere in a file, or modification of an existing block, or deletion of any block. The proposed solution is based on a new variant of authenticated dictionaries, where rank information is used to organize dictionary entries. Thus efficient authenticated operations on files at the block level, such as authenticated insert and delete can be supported. The security of proposed constructions is proved using standard assumptions.

'Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing' [8] is a work that deals with security of data stored in cloud. It is distributed in 2 phases. In phase 1 the Owner calculate the MAC on each partitioned file block going to be stored in a cloud server. Then the file blocks are transferred cloud server and the key is shared with the Third Party Auditor (TPA). At the time of confirmation auditing phase, the TPA requests from the cloud server a number of randomly selected blocks and their corresponding MACs to verify the correctness of the data file. This scheme has a major drawback i.e. if TPA is not trustworthy then data may lead to outside world.

Another paper that explored the problem of public verifiability in Cloud is 'Privacy-Preserving Public Auditing for Secure Cloud Storage' [9]. In the scheme proposed in this paper, a public key is used to identify a trusted third party. This means that any party who is in possession of this public key is considered as a trusted Third Party Auditor (TPA). Here it is assumed that a TPA is unbiased while the server is untrusted. This scheme has crucial differences from that of the existing PDP models in the verification process. These schemes do not consider dynamic data operations, and the block insertion cannot be supported at all. This is because the construction of the signatures is involved with the file index information. Therefore, once a file block is inserted, the computation overhead is unacceptable since the signatures of all the following file blocks should be re-computed with the new indexes. To deal with this limitation, the index information in the computation of signatures is removed so that the individual data operation on any file block will not affect the others.

## VI. AN AUDITING PROTOCOL FRAMEWORK USING BILINEAR PAIRING

Many of the auditing protocols could be subjected to some attacks. The most important among them is the replay attack. Here the attacker which is the untrusted Server, can save the proof for the challenges issued by an auditor or a Client and then reuse these proofs to assure the Auditor or Client that the data is secure. Some existing remote integrity checking methods can only serve for static archive data and, thus, cannot be applied to the auditing service since the data in the cloud can be dynamically updated. Thus, an efficient and secure dynamic auditing protocol is desired to convince data owners that the data are correctly stored in the cloud.

Use of mathematical concepts can ensure the security of data much better than normal cryptographic methods. Furthermore it is required that the mathematical process in use should be such that there is little or no use of the original content. In [1] first an auditing framework for cloud storage systems in designed and an efficient and privacy-preserving auditing protocol is proposed which is based on the property of bilinear pairing. Bilinear pairing is a strong mathematical function that can be used for integrity checking without requiring the original data content. Then, we extend the auditing protocol to support the data dynamic operations, which is efficient and provably secure. The auditing

protocol is further extended to support batch auditing for both multiple owners and multiple clouds, without using any trusted organizer.

## VII. CONCLUSION

This paper provided a brief introduction into the emergence and importance of storage services in Cloud computing. The paper also discusses issues in the area of data security in Cloud and provides a general description of major auditing protocols used in the area of Cloud.

## VIII. ACKNOWLEDGMENT

The authors would like to thank the staff and students of Computer Science and Engineering Department and the management of Viswajyothi College of Engineering And Technology, Vazhakulam, India for their continued support and guidance.

## IX. REFERENCES

[1] Peter Mell, Timothy Grance "The NIST Definition Of Cloud Computing", September 2011.

[2] Data Storage Services In Cloud – Internet url:http:// computer.howstuffworks.com/cloud-computing/cloud-storage1.htm"

[3] Yanpei Chen, Vern Paxson, Randy H. Katz, "What's New About Cloud Computing Security?", Electrical Technical Report No. UCB/EECS-2010-5

[4] Cong Wang and Kui Ren,Wenjing Lou, Jin Li, "Toward Publicly Auditable Secure Cloud Data Storage Services" , IEEE Network 2010

[5] Giuseppe Ateniese, Randal Burns, Reza Curtmola, Joseph Herring, Lea Kissner, Zachary Peterson and Dawn Song "Provable Data Possession at Untrusted Stores", Proc. of ACM Conf. 2007

[6] Giuseppe Ateniese, Roberto Di Pietro, Luigi V. Mancini, and Gene Tsudik, "Scalable and Efficient Provable Data Possession", IACR Cryptology archive ,2008

[7] C Chris Erway, Alptekin K, Charalampos Papamanthou and Roberto Tamassia "Dynamic Provable Data Posession", Proc. ACM Conf. Computer and Communications, 2009.

[8] Qian Wang,Cong Wang, Kui Ren,Wenjing Lou, and Jin Li "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing", IEEE Transactions on Parallel and Distributed Systems, 2011

[9] S M Chow,Qian Wang,Cong Wang, Kui Ren and Wenjing Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage", Proc. ACM Symp. Applied Computing,2011

[10] K.Yang and X.Jia "An Efficient and Secure Dynamic Auditing Protocol for Data Storage in Cloud Computing" , IEEE transaction on parallel and distributed systems, 2013.

[11] Ran Canetti and Ron Rivest,"Pairing-Based Cryptography" , Special Topics in Cryptography, 2004.