IJASRM International Journal of Advanced Scientific Research and Management, Vol. 1 Issue 6, June 2016.
www.ijasrm.com
ISSN 2455-6378

A Novel Approach to Enhance Security in Cloud

Chandan Reddy[1], Dipin Raj[2] , Sachin K.S.[3], Venugopal[4]and Vaidehi M[5]

[1,2,3,4,5] Department of Information Science and Engineering,
Dayananda Sagar College of Engineering,
Vishveswara Technological University,
Bangalore, India

Abstract

The exponential growth in Cloud technology has driven organizations to outsource data from local to remote cloud storages. Most of the infrastructure service providers like, Microsoft, Amazon and Google are providing services were in their clients can easily store and access data. But the concern in such scenario is about data Security and Integrity. To address these technical challenges, this paper presents a model to enhance data integrity and data deduplication in Cloud. Here we present two secure models, SecCloud and SecCloud +. The ScCloud does data assessment with maintenance of a MapReduce cloud, which helps the users to create data tags before uploading as well assess the integrity of data that has been stored in the remote server. The SecCloud + is another model used in this system which enables the clients to encrypt their data before uploading to the remote server, this approach is to enhance data integrity.

Keywords: *Cloud, SecCloud, SecCloud^{+},Data Security*

1. Introduction

Cloud being one of the most successful storage and computing models, were the clients store their data into the cloud using advanced communication networks, servers and avail computing resources from shared pool of highly configurable resources. The advantage of the cloud model is that, it is highly flexible easy to maintain and charged as per the usage. This strategy which is highly economic has encouraged the clients to incline towards the Cloud model. This model which also supports multitenancy has an issue of Security and Data Integrity.

This sometimes de- motivates the Clients to use the Cloud model due to unreliable data security. This model is more susceptible to data hacking and data corrupting.

In recent years, more and more events on cloud service outage or server corruption with prominent infrastructure service providers are reported [1-3,4]. At times there is a possibility of data breach. The Cloud service providers should ensure their clients of providing strong security, data privacy and reliability, else it would be difficult for the clients to avail cloud services based on economic savings and flexibility of their services.

Most of the Cloud Infrastructure Service providers enhance security through modification of the existing Encryption and Decryption techniques.

To concentrate on these issues, in this paper we propose two models. The SecCloud model which ensures data security, and other model SecCloud+ which provides data integrity. Here we aim at employing a data encryption technique to avoid data corruption and enhance data integrity. In this paper Section 2 represents the Literature Survey, Section 3 presents the Problem definition, Section 4 presents the Methodology, Section 5 presents Result Analysis and snapshots and in section 6 the conclusion is presented and finally the References..

2. Literature Survey

As there is enormous growth and advancement in the Cloud environment, few critical issues like Security, data Integrity are arising. In recent years most of the researchers are focusing towards enhancing the

performance of the Cloud model by improving the aforementioned.

J. Yuan et. al., emphasise on Data integrity and Storage. Their work focuses on Proof of Retrievability (POR) and Proof of Data Possession (PDP) techniques which assure data integrity for cloud storage. Proof of Ownership (POW) improves storage efficiency by securely removing unnecessarily duplicated data on the storage server [4]. To resolve these issues, the authors have proposed few efficient techniques which include polynomial-based authentication tags and homomorphic linear authenticators. Numerical analysis and experimental results on Amazon AWS show that their scheme is efficient and scalable.

S. Halevi, et. al., have worked towards data deduplication. In their work the authors have identified attacks that exploit client-side deduplication, allowing an attacker to gain access to potentially huge files of other users based on a very small amount of information. To overcome such attacks, the authors have introduced Proofs-of-ownership (PoWs), where a client proves to the server that it actually holds the data of the file and not just some short information about it. They have formalized Proof-of-ownership and have presented solutions based on Merkle trees and specific encodings, and analyze security [5].

Researchers G. Ateniese et.al. have introduced a model for provable data possession (PDP) that can be used for remote data checking. Their proposed 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.

Experiments using their implementation verify the practicality of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation. They have also conducted an in-depth experimental evaluation to study the tradeoffs in performance, security, and space overheads when adding robustness to a remote data checking scheme [2].

F. Seb´e, J et. al., have worked on remote data possession checking protocols. Yhese permit to check that a remote server can access an uncorrupted file in such a way that the verifier does not need to know beforehand the entire file that is being verified. Unfortunately, current protocols only allow a limited number of successive verifications or are impractical from the computational point of view.

In their work, the authors have presented a new remote data possession checking protocol such that it allows an unlimited number of file integrity verifications and also its maximum running time can be chosen at set-up time and traded off against storage at the verifier [7].

3. Problem Definition

Even though cloud storage system has been widely adopted, it fails to accommodate some important emerging needs such as the abilities of auditing integrity of cloud files by cloud clients and detecting duplicated files by cloud servers.

Existing System

The Cloud server has enabled the users to store and manage data efficiently. Data is transferred through internet and stored in an uncertain domain, not under control of the clients at all, which inevitably raises clients great concerns on the integrity of their data. These concerns originate from the fact that the cloud storage is susceptible to security threats from both outside and inside of the cloud, and the uncontrolled cloud servers may passively hide some data loss incidents from the clients to maintain their reputation. There are possibilities that the cloud server may discard few files belonging to the clients due to the resource constrain.

The rapid adoption of cloud services is accompanied by increasing volumes of data stored at remote cloud servers. Among these remote stored files, most of them are duplicated: according to a recent survey by EMC , 75% of recent digital data is duplicated copies. This fact raises a technology namely deduplication, in which the cloud servers would like to deduplicate by keeping only a single copy for each file (or block) and make a link to the file (or block) for every client who owns or asks to store the same file (or block). Unfortunately, this action of deduplication would lead to a number of threats potentially affecting the storage system.

The drawbacks of the system are

- The Clients cannot efficiently perform periodic integrity check without the local copy of data files.
- The Cloud server will not efficiently confirm that the client owns the uploaded file prior to creating a link to the file.

IJASRM
International Journal of Advanced Scientific Research and Management, Vol. 1 Issue 6, June 2016.
www.ijasrm.com
ISSN 2455-6378

4. Methodology

In this paper, we are aiming at achieving data integrity and deduplication in cloud, we propose two secure systems namely SecCloud and SecCloud+.
SecCloud introduces an auditing entity with maintenance of a MapReduce cloud, which helps clients generate data tags before uploading as well as audit the integrity of data having been stored in cloud. In addition, SecCoud enables secure deduplication through introducing a Proof of Ownership protocol and preventing the leakage of side channel information in data deduplication. SecCloud+ is an advanced construction motivated by the fact that customers always want to encrypt their data before uploading, and allows for integrity auditing and secure deduplication directly on encrypted data.

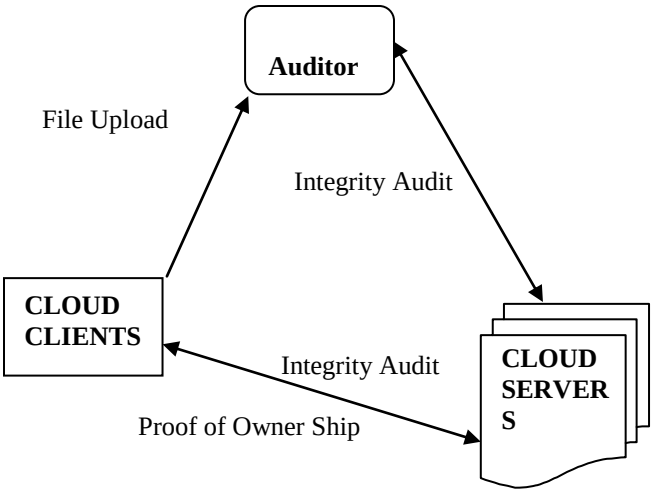

Figure 1. Data Security and Integrity Flow

In this paper we have used simple key generation algorithms and tag generation algorithm to achieve the same. The figure 1 shows that the data to be uploaded by the Client is audited by the Auditor. Later the same file is transferred to the Cloud Servers for storage. At the time of access by the Clients, the Proof of Ownership is verified and then forwarded to the Client.

In the SecCloud system, there are three entities:

- Cloud Clients-Clients have large data file to be stored and retrieved
- Cloud Servers- Virtualizes the resources according to the requirements of clients and expose them as storage
- Auditor- Helps clients upload and audit their outsourced data

The following actions are performed by the proposed system

- File Uploading Protocol
- Integrity Auditing Protocol
- Proof of Ownership Protocol
- File Uploading Protocol
- This protocol aims at allowing clients to upload files via the auditor
- Integrity Auditing Protocol
- Plays a vital role in providing Integrity verification
- Proof of Ownership Protocol
- Verifies that the client exactly owns a claimed file

5. Results

Through this proposed approach it is observed that both data integrity and deduplication in cloud has been achieved. Figure 2 shows few samples which depicts the enhancement in data security and integrity

Fig 2: Pages depicting the User verification prior to data access from the Cloud

6. Conclusions

Aiming at achieving both data integrity and deduplication in cloud, we propose SecCloud and SecCloud+. SecCloud introduces an auditing entity with maintenance of a MapReduce cloud, which helps clients generate data tags before uploading as well as audit the integrity of data having been stored in cloud. In addition, SecCoud enables secure deduplication through introducing a Proof of Ownership protocol and preventing the leakage of side channel information in data deduplication. Compared with previous work, the computation by user in SecCloud is greatly reduced during the file uploading and auditing phases. SecCloud+ is an advanced construction motivated by the fact that customers always want to encrypt their data before uploading, and allows for integrity auditing and secure deduplication directly on encrypted data..

References

[1] J. Li, X. Chen, M. Li, J. Li, P. Lee, and W. Lou, "Secure deduplication with efficient and reliable convergent key management," IEEE Transactions on Parallel and Distributed Systems, vol. 25(6), 1615–1625,(2014).

[2] R. Di Pietro and A. Sorniotti, "Boosting efficiency and security in proof of ownership for deduplication," in Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security, ser. ASIACCS '12. New York, NY, USA: ACM, 81–82,(2012).

[3] J. Li, X. Tan, X. Chen, and D. Wong, "An efficient proof of retrievability with public auditing in cloud computing," in 5th International Conference on Intelligent Networking and Collaborative Systems (INCoS), 93–98,(2013).

[4] W. K. Ng, Y. Wen, and H. Zhu, "Private data deduplication protocols in cloud storage," in Proceedings of the 27th Annual ACM Symposium on Applied Computing, ser. SAC '12. New York, NY, USA: ACM, 441–446,(2012).

[5] J. Douceur, A. Adya, W. Bolosky, P. Simon, and M. Theimer, "Reclaiming space from duplicate files in a Server less distributed file system," in 22nd International Conference on Distributed Computing Systems, 617–624,(2002).

[6] M. Bellare, S. Keelveedhi, and T. Ristenpart, "Message-locked encryption and secure deduplication," in Advances in Cryptology – EUROCRYPT 2013, ser. Lecture Notes in Computer Science, T. Johansson and P. Nguyen, Eds. Springer Berlin Heidelberg,vol. 7881, 296–312(2013).