Authorized Public Auditing of Dynamic Storage on Cloud with Efficient Verifiable Fine-Grained Updates with MHT

Mayur Lokhande
Department of Information Technology
S.P. Pune University, Pune, India

Supriya Kapse
Department of Information Technology
S.P. Pune University, Pune, India

Sonali Darekar
Department of Information Technology
S.P. Pune University, Pune, India

Mahesh Sonawane
Department of Information Technology
S.P. Pune University, Pune, India

Abstract

One of the top technology concept is Cloud Computing. Cloud storage servers plays an important role in the technology buzz – cloud computing where clients can store their data at cloud servers and can access this data from anywhere and anytime. This opens up the threat on the data which client is storing on the cloud as it has to stored and accessed via Internet from any device. Here we came up with data Security which opens up lot of areas like authentication of clients, data integrity checks and data privacy preserving. In our proposed work we are going to focus on data integrity area which is concerned about data security. When we consider about data security data auditing is concept of frequently auditing the integrity of the data to make sure that the data is not tampered and it is secured. Most of the times clients offload the auditing process of their data to Third Party Auditors (TPA) who are able to check the validity of the data without actually accessing the data on cloud. When the verification is done by a trusted third party called data auditing in which third party is called an auditor. There are several drawbacks of these techniques. One of the issue arises here is a necessary authentication process is missing between the auditor and cloud service provider it means that anyone can challenge the cloud service provider for a proof of integrity of certain file which puts auditing service at risk. In our work, we provide a legal Study for possible types of fine-grained data updates and suggest a scheme that can fully support authorized auditing and fine-grained update request.

Keywords- Cloud computing, Data security, authorized public auditing, fine-grained updates

I. INTRODUCTION

Cloud computing is a new technology of computing in discriminate to original desktop computing.

Today, this new concept become popular and provided more concentration. Cloud computing is useful remotely so user outsource the data on cloud and view remotely anywhere.

Cloud is new network technology over the internet is used to describe a new class. Cloud computing environment provides useful services to be met are infrastructure and service providers. The Cloud platforms are managed and hire out the resort to users on-demand, by the infrastructure providers although current development and growth of cloud computing is fast, a discussion, and vacillation on the usage of cloud still live. Data risk/secrecy is one of the most important in the adoption of cloud computing Compared to customary systems, users will lose their direct control over their data. In remote method there is no surety of the integrity.

In our suggested work we will introduce the problem of integrity to prove that something exist is true or to make certain that something is correct big data storage on cloud. This verification conducted by the TPA (trusted third party) so TPA is the auditor. From cloud users mental view, it is named as auditing-as-a-service. In a remote that one or more reason or believing that something is or is not true or confirms the accuracy or truth of something scheme, the cloud storage server (CSS) cannot equip effective virtue evidence of a given part of data to a verifier unless all this data is broken. Integrity checking is very important task in cloud .to check integrity lost or preserve, if lost the integrity that means there is changes in the data, if not then there no any changes in the data on cloud. So that Data auditing is an important part in the cloud and auditing should be trusted user i.e. TPA.

A. Problem Statement

Cloud environment have number of advantages, providing infrastructure as a service and maintenance as a service. Cloud work remotely so security is major factor relieves the burden of user's task but security became a major concern in all time. TPA is auditor so user r hires a TPA to check the integrity of data stored in cloud server. But again the problem arises whether the TPA is authorized or not. Another concern is related to the utilization of resources in cloud environment. There are number of resources as well as requests. There is no better way to serve the requests within a particular time and with available resource.

Previously scheduling algorithms were performed in grid but reduces the performance by requiring advance reservation of resources. In cloud environment due to scalability of resources, manually allocate resources to task is not possible.

II. LITERATURE SURVEY

In past there have been lot of work has been done on cloud data security different techniques were used to provide security to cloud data but there are some disadvantages of such systems. Existing methods for protecting user data include data encryption prior to storage and user authentication procedures prior to storage or retrieval of data after that building secure channels for data transmission over the cloud. In this existing systems the algorithms used are cryptographic and Digital signature based.

First work is by Ateniese et al who consider public auditability in provable data possession model for ensuring possession of files on untrusted storages. Ateniese present a model in which RSA based homomorphic tags are used. With the help of this technique public auditability concept is achieved. But the problem with this model is that it does not support dynamic data operation and also suffer security problems [8]. Another research by wang considered dynamic data storage in a distributed scenario which is a better idea. He proposed challenge response protocol can both determine the data correctness and locate possible errors but this model only considered partial support for dynamic data operations [9].

Kaliski presented a proof of retrievability model. The main disadvantage of this model as it does not support public auditability. Extended research on this done by Shacham and Waters design an improved PoR scheme with full proofs of security in the security model. In this model they use publicly verifiable homomorphic authenticators built from BLS signatures based on which the proofs can be aggregated into a small authenticator value by using this public retrievability is achieved. The main concern comes in front with this is the authors only consider static data files which are not preferable because our main concern is about big data files [10].

One research was there on MAC based scheme which has the disadvantages like the number of times a particular data file can be audited is limited by the number of secret keys that must be fixed a priori. So the problem arises here is once all possible secret keys are exhausted after that the user then has to retrieve data in full to recomputed and republish new MACs to TPA[10] .Here in this scheme TPA also has to maintain and update state between audits that is to keep track on the revealed MAC keys. It can only support static data and cannot efficiently deal with dynamic data at all so this is a big issue to be solved when considering big data.

HLA based scheme- There is need of system which can verify integrity of data without retrieving data blocks present. So there is another method presented that is HLA scheme was used for this purpose. The only difference between HLA and MAC is that HLA can be aggregated. The main issue with this system is that data can be retrieved only if linear combinations of same block is used [10].

III. SYSTEM IMPLEMENTATION

The brief Explanation of the System Architecture will clear the all points.

A. Authentication

The Cloud Service users, like File Owner and Third Party Auditor are authenticated by the Cloud Service provider. Even after registering with the application the user is not allowed to access any part of application until the user is qualified by the CSP. Users need to be activated by the CSP, and similarly can be deactivated too.

The TPA is responsible for checking the integrity of the file and report the file owner about status. So the TPA is also authenticated by the file owner to access the file.

Additionally, even the TPA is accessing the file for checking the integrity but actually TPA will get the illusion of accessing the file but there is nothing like file that TPA is accessing. The Values on which TPA finds the result of integrity is nothing but the Hash values of parts of encrypted files computed collectively. So the actual original file is not present on the server as it is.

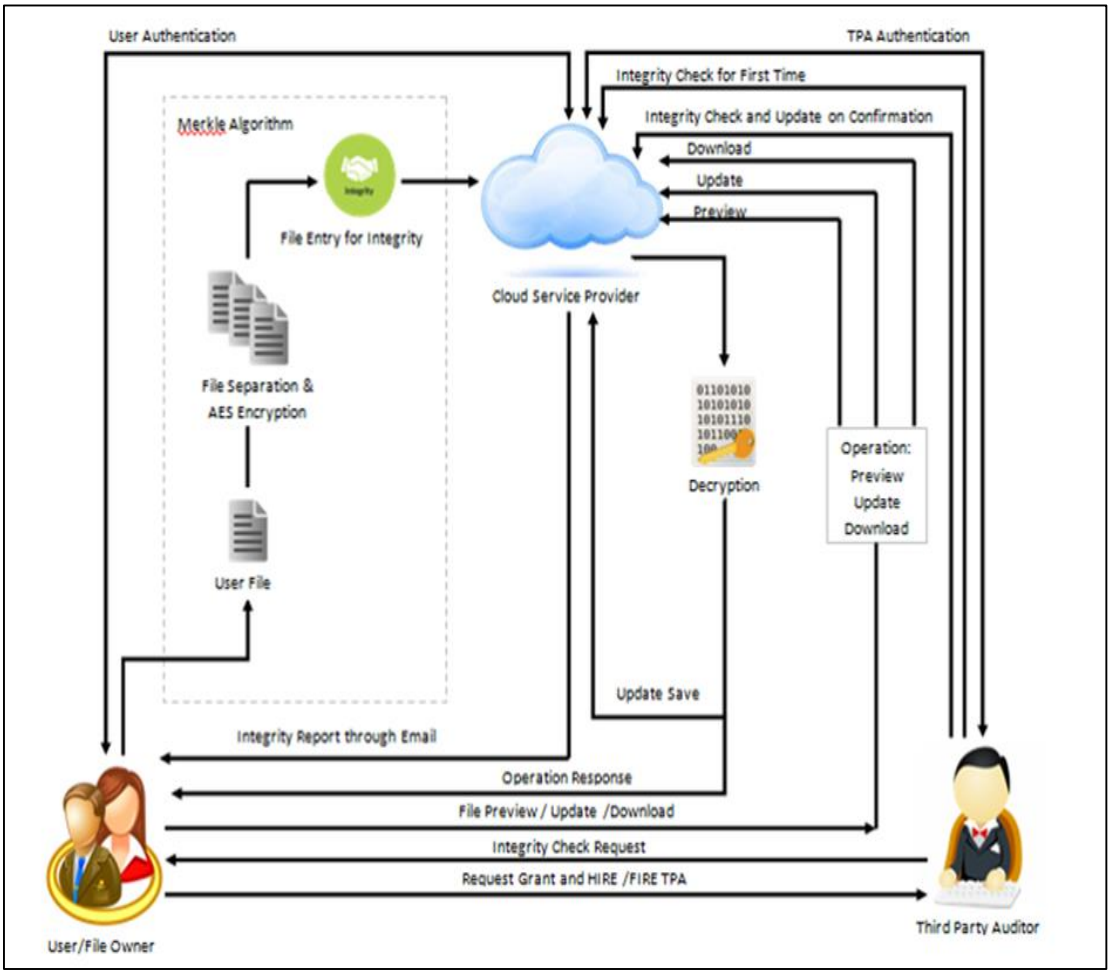

Fig. 1: System Architecture

B. File Uploads

The user uploads the files on CSP space. The files are portioned in the parts which are the encrypted with the 256bit AES algorithm. These files are used by the SHA1 algorithm to compute its hash key. If any changes done by the user in any part will reflect in hash value for the whole file. But in realty the whole file in actual format is not present on the service space.

This constitutes the Merkle Hash Tree Algorithm which is the main part of our project.

C. Integrity Check

When user uploads the files that need to be checked by the Hired TPA. This will be done by the Integrity check functionality.

- First time integrity check
- Integrity Preserved.
- Integrity Lost.

This are the possible outcomes of the integrity check by the TPA.

D. File Operation

1) File Update

File Updating is normal function of the user. But the innovative idea here is that user is updating only a required part and the part is in encrypted format. After updating the part will again be encrypted in its location and preserves its integrity. The Encryption and decryption needed to do the work.

2) File Preview

File Preview also need file to encrypt before previewing.

3) File Download

File needs to be in decrypted before downloading. Here the AES Decryption run for particular file of Particular User. The File will be downloaded.

4) Request for Integrity Check

The third party Auditor is nothing but the company which will provides the service of checking and reporting the integrity status of the user. The Third Party auditor will see the list of cloud service user. The TPA will send the request to the file owner to hire me as a TPA for your documents.

Then the user will decide whether to Grant or ignore the request. This will prevent the file owner to send request to fraudulent TPA to check the files. By this way the file owner will be in key place to allow TPA for integrity check or not.

Additionally, if after some time file owner thinks that he/ she don't need the TPA to check files integrity anymore. He / She can HIRE or FIRE the TPA.

5) Priority Based Scheduling

As the resource may be shared by the multiple entities at may be same time or different time, so there should be the mechanism for allocating resource.

As JSP Servlets is already a thread safe which will allot the separate instance of the object to the requester. Additionally, we added the Priority based Algorithm to maintain its integrity.

The higher priority gives to the File owner tasks, like file Updating.

The lower priority given to TPA task such as, checking the integrity of the file. The TPA is not allowed to get access to the file when the user is updating the file.

IV. PROPOSED SYSTEM

The various main steps of our proposed scheme is described as follows:

1) The client will generate keying materials via KeyGen and FileProc after that he upload the data to CSS. Different from previous schemes here in our scheme the client will store a RMHT as metadata.
2) After that the client will authorize the TPA by sharing a value sigAUTH.
3) Verifiable Data Updating: the CSS performs the client's fine-grained update requests via Perform Update on user's data.
4) Client runs Verify Update to check whether CSS has performed the updates on both the data blocks and their corresponding authenticators (used for auditing) honestly.
5) Challenge, Proof Generation and Verification: Describes how the integrity of the data stored on CSS is verified by TPA via GenChallenge, GenProof and Verify.

Auditability aware data scheduling: In this scheme we are clustering various tasks submitted in an application both from the user and auditor on the basis of their priority.

Various phases of our proposed work are as follows.

A. Key Generation Algorithm

1) Start
2) Read Data owner id (udoid)
3) If(doid==udoid) (Execute step 4 to 10)
4) Read secret key (ssk)
5) Choose random number α from Zp(α <- Zp)
6) Choose random group generator (g) from Zp
7) Calculate v=gα
8) Display secrete key pair spk=(v, ssk)
9) And Public key pair spk=(v,spk)
10) Update α value on alpha xml in sky drive.
11) Stop
12) Else
13) Stop.

B. Signature Generation Algorithm

1) Start
2) Read data owner id (udoid)
3) If (doid ≠ udoid)
4) Stop
5) Else
6) Read file path (Fp)
7) Read No. of levels (n) for the construction of MHT
8) Calculate the block size of MHT =size of file/n.
9) Divide the file into NOB Blocks
10) For i=0;
11) For (i<=0) && (i>=NOB)

12) Calculate Hc[i] = enceyptsha1 [block[i]]
13) Display hc[i]
14) Choose random number u from set of group generators 'G'
15) if(i<=0 &&i>=NOB)
16) Calculate Sig[i] = (hc[i]*) α
17) Display sig [i]
18) Construct MHT and generate Root node(R)
19) Generate signature for root node rootsign = H(R))α
20) Upload file to web server
21) Update hash values & signature on TPA xml on Sky Drive.

C. Data Integrity Verification by TPA Algorithm

1) Start
2) Read data owner id(udoid)
3) If (doid ≠ udoid)
4) Stop
5) Read file nam from AWS
6) Retrieve No. of blokes from TPA xml
7) Select the blocks number the user want to verify.
8) Get the auxiliary information for block chal from TPA xml
9) Based on Auxiliary information generate new root for MHT
10) If (new root ≠ root) file modified
11) Else File not modified
12) Stop.

V. CONCLUSION

In today's world, important aspect for cloud user is cloud data security and privacy. How to ensure trusting a third party we present an overview in this paper. TPA cannot derive user's data during the process of public data auditing because it focused on privacy-preserving for datasets. The proposed system is such that it will prevent malicious TPA to make an illegal copy of data, which uses a better signature scheme. For increases the efficiency of update process it provides a feature of fine-grained dynamic data update. This paper proposes an algorithm based on priority which will schedule the tasks coming to CSP.

REFERENCES

[1] Yuri Demchenko, Cees de Laat. Defining Architecture Components of the Big Data Ecosystem.
[2] D. Zissis and D. Lekkas, ''Addressing Cloud Computing Security Issues,'' Future Gen. Comput. Syst., vol. 28, no. 3, pp. 583-592, Mar. 2011.
[3] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, ''Scalable and Efficient Provable Data Possession,'' in Proc. 4th Int'l Conf. Security and Privacy in Commun. Netw. (SecureComm),2008, pp. 1-10.
[4] 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. Inf. Syst. Security, vol. 14, no. 1, May 2011, Article 12.
[5] G.Ateniese, R.B. Johns,R. Curtmola, J.Herring, L. Kissner,Z. Peterson and D. Song, ''Provable Data Possession at Untrusted Stores,'' in Proc. 14th ACM Conf. on Comput. and Commun. Security (CCS), 2007, pp. 598-609.
[6] A. Juels and B.S. Kaliski Jr., ''PORs: Proofs of Retrievability for Large Files,'' in Proc. 14th ACM Conf. on Comput. and Commun. Security (CCS), 2007, pp. 584-597.
[7] H. Shacham and B. Waters, ''Compact Proofs of Retrievability,'' in Proc. 14th Int'l Conf. on Theory and Appl. of Cryptol. And Inf. Security (ASIACRYPT), 2008, pp. 90-107.
[8] S. Nepal, S. Chen, J. Yao, and D. Thilakanathan, ''DIaaS: Data Integrity as a Service in the Cloud,'' in Proc. 4th Int'l Conf. on Cloud Computing (IEEE CLOUD), 2011, pp. 308-315.
[9] Y. He, S. Barman, and J.F. Naughton, ''Preventing Equivalence Attacks in Updated, Anonymized Data,'' in Proc. 27th IEEE Int'l Conf. on Data Engineering (ICDE), 2011, pp. 529-540.
[10] E. Naone, ''What Twitter Learns from All Those Tweets,'' in Technology Review, Sept. 2010, accessed on: March 25, 2013. [Online]. Available: http://www.technologyreview.com/view/420968/what-twitter-learns-from-all-those tweets/
[11] X. Zhang, L.T. Yang, C. Liu, and J. Chen, ''A Scalable Two-Phase Top-Down Specialization Approach for Data Anonymization Using MapReduce on Cloud,'' IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 2, pp. 363-373, Feb. 2014.
[12] S.E. Schmidt, ''Security and Privacy in the AWS Cloud,'' presented at the Presentation Amazon Summit Australia, Sydney, Australia, May 2012, accessed on: March 25, 2013. [Online]. Available: http://aws.amazon.com/apac/awssummit-au/.