Developing Third Party Auditing Scheme for Secure Cloud Storage Service

Manasi Doshi
PG student of Department of Computer Engineering,SCOE, Vadgaon, Pune-411048

Swapnaja Hiray
Associate Professor of Department of Computer Engineering,SCOE, Vadgaon, Pune-411048

ABSTRACT
Cloud computing is the use of computing of sources that are delivered as a service over a network. Cloud enables users to store their data, but data is stored at remote location. A major characteristic of the cloud services is that user's data are usually processed remotely in unknown machines that users do not operate. So, basic need is to provide security to cloud server. To achieve this, perform flexible distributed storage, utilizing the homomorphism token and distributed erasure-coded data. Also allows for strong cloud storage correctness and simultaneously achieves fast data error localization.

Keywords
Cloud computing, security, distributed storage, error localization

1. INTRODUCTION
In cloud computing we can share our data and application at common place. This uses internet and share resources to provide services. Security is important issue because cloud having many benefits so, it have many users. This paper focuses towards security to cloud.

It is based on distributed storage on 3 machines. It uses homomorphic token for checking integrity of data. This helps user low cost communication and computational cost. The auditing result ensures strong cloud storage correctness as well as simultaneously achieves fast data error localization, i.e., the identification of misbehaving server. It allows client to perform secure and efficient dynamic operations on outsourced data, including block modification, deletion, and append [1].

2. RELATED WORK
In this section we first review related works addressing security in cloud. Security issue is very important in cloud there are many techniques available so here is review of all these.

Data security is the major challenge in the cloud computing as user's data reside in the servers which are remotely situated and far away from the end-users. These data may include confidential data (financial data, health records), personal information which may be disclosed to competitors or publicly. So security emerges as the highest priority issue [2]. In [3] Third party auditor for verification, they describes three network entities i.e. client which is user, cloud storage server which is handled by cloud service provider and Third party auditor which is verifier.TPA having public key, it is act with only trusted server, they are not focuses on data privacy. In [4] it defines 2 basic schemes. Scheme 1 : User computes the MAC of every file block. Transfers the file blocks & codes to cloud and shares the key with TPA. During the Audit 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. Drawback of this scheme is TPA can see cloud data. Scheme 2: In Setup phase, User uses s keys and computes the MAC for blocks and user shares the keys and MACs with TPA. During Audit, TPA gives a key (one of the s keys) to CSP and requests MACs for the blocks. TPA compares with the MACs at the TPA. Improvement from Scheme 1: TPA doesn't see the data, preserves privacy. Drawback: a key can be used once, Schemes 1 & 2 are good for static data (data doesn't change at the cloud). In paper [5] they discuss main challenges for achieving cloud computing services, this problem focuses on accountability in cloud computing. Accountability means verification of access control policies.

3. PROPOSED WORK
3.1 System Modules:
1. Client:
Client is that entity who is using of cloud services and who has to store data on cloud. Multiple clients can use cloud storage services.

2. TPA:
TPA is an optional entity. It has expertise and capability to expose dummy client. E.g. authentication of client.

3. CSP:
CSP is an entity which provides cloud services. E.g. client want to upload file then CSP give call to CS.

4. CS:
CS is an entity which allow client to perform operation on data stored on itself.

5. Main Backup server:
It is an entity which stores complete file.

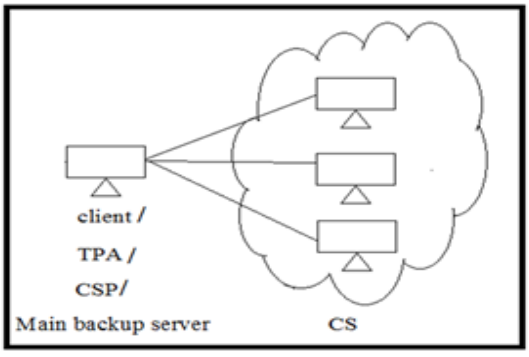
3.2 System Architecture


Figure 1. Proposed cloud storage service architecture (using 4 machines)

Here client, CSP, TPA and main backup server working on single machine. For storing client's data i.e. input file is

divided into 3 blocks and are store on CS. Each block on 1 machine and these blocks are formed depending on file size. Size of each block is same.

3.3 Ensuring cloud data storage

3.3.1 Storage of data in a Cloud

In a cloud, the users are expected to store their data in a cloud and once stored, the data is not accessed locally. As a result it is important to maintain the correctness and availability of the data files being stored on the distributed cloud servers must be guaranteed. Hence, it is challenging to find out the unauthorized or corruption of data on servers and also more importantly in a distributed case when such inconsistencies are successfully detected. It is also equally challenging to find the server on which the error occurs as it would help resolve the issue faster. We will look at the method to overcome this issue in various parts. To start with, the basic tools will be reviewed and then homomorphic tokens are introduced. Moving on, we will look at challenge response protocol for verifying the storage correctness as well as identifying misbehaving servers and will conclude with the procedure for file retrieval and error recovery based on erasure-correcting code.

3.3.2 File Distribution Preparation

File which client want to store is divides into 3 blocks depending upon size of original file. Each new block is of same size. To achieve availability each block is stored twice. E.g. 1st block is stored on CS 1 and CS 2, 2nd block is stored on CS 2 and CS 3, 3rd block is stored on CS 3 and CS 1.

3.3.3 Token Pre-computation

To ensure that the data present on the cloud is correct, the method we are looking at depends on the pre-computed verification tokens as the main idea is that the user pre-computes a certain number of short verification token before file distribution. Towards end when the users want to ensure data storage correctness, they confront the servers of the cloud with a lot block indices that are generated in an arbitrary manner and when these confrontations are received, each cloud server computes a short signature over the specified blocks and returns them to the user.

3.3.4 Verification and Error Localization

The main requirement for getting rid of errors in the storage system is error localization. Though many previous methods do not consider the data error localization as an issue and only provide binary results for the storage verification.

4. ALGORITHMS

3.4 4.1 Token Generation

Procedure:

1. Sum=0
2. number of tokens t=3
3. size of file= m; each part having size=n
4. number of indices=r=8;
5. for t← 1, 3 do
6. for round← 1, n do
7. for r← 1,8 do
8. integer c=get value present at round[r];
9. integer d= power (r, c)
10. sum=sum + d;
11. end for
12. end for
13. end for
14. store in database

end procedure

3.5 4.2 Error Localization

1. Challenge for particular file
2. Get file parts from 3 servers of that file
3. Check all file part is present or not
4. Get complete file from backup server.
5. Divide that backup file into three parts
6. Compare each file part from main serve with each file part from backup server.
7. If bytes are not matched then it returns which server is misbehaving.

3.6 4.3 Error Recovery

1. Once we identify which file part from which server is misbehaving.
2. Get that file parts from backup server.
3. Once we get place that file part into main server.
4. Now when we download file we get complete our original file.

5. CONCLUSION

From all above discussion, it shows that we can provide security to data stored on cloud i.e. providing security to remotely stored data is possible. With the help of tokens generation and token matching we are providing security. By taking backup of data we can achieve availability even if CS crash. It allows user to perform block operation i.e. append, delete, modify as well as to give challenge to uploaded to check correctness of data. In future focus will be towards performance, CPU utilization etc.

6. ACKNOWLEDGEMENT

I am very thankful to my guide for guiding me and heartly thankful to IJCA to give me such a wonderful chance for publishing my paper.

7. REFERENCES

[1] C. Wang, Q. Wang, K. Ren, N. Cao, W.Lou "Toward Secure and Dependable Storage Services in Cloud Computing" IEEE transactions on services computing, VOL. 5, NO. 2, APRIL-JUNE 2012

[2] C. Deyan and Z. Hong, "Data Security and Privacy Protection Issues in Cloud Computing," in International Conference on Computer Science and Electronics Engineering (ICCSEE), 2012 pp. 647- 651.

[3] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing," Proc. 14th European Conf. Research in Computer Security (ESORICS '09), pp. 355-370, 2009.

[4] C. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy- Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, preprint, 2012, doi:10.1109/TC.2011.245.

[5] Ryan K L Ko, Peter Jagadpramana, Miranda Mowbray, Siani Pearson, Markus Kirchberg, Qianhui Liang and Bu Sung Lee,"TrustCloud: A Framework for Accountability and Trust in Cloud Computing," 2nd IEEE Cloud Forum for Practitioners (IEEE ICFP 2011), Washington DC, USA,pp 1-8.

[6] C. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy- Preserving Public Auditing for Secure Cloud

Storage," IEEE Trans. Computers, preprint, 2012, doi:10.1109/TC.2011.245.

[7] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing," IEEE Trans. Parallel and Distributed Systems, vol 22,no. 5, pp. 847-859, 2011.

[8] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling Public Verifiability and Data Dynamics for Storage Security in Cloud Computing," Proc. 14th European Conf. Research in Computer Security (ESORICS '09), pp. 355-370, 2009.

[9] Amazon Cloud, http://aws.amazon.com