Security and Verification of Data in Multi-Cloud Storage with Provable Data Possession

Vrushali K Gaikwad
PG Student
Computer Engineering Department,
DYPSOET Pune-43(India)

Ramesh Kagalkar
Asst.Proffessor
Computer Engineering Department,
DYPSOET Pune-43(India)

ABSTRACT
Data integrity verification is one of the biggest security issue in cloud. To check integrity of data Provable data possession is one of the method available. In this paper, we have created an efficient PDP method for distributed cloud storage, in which we multiple cloud service providers are maintaining and storing client's data in cooperative way. This cooperatively working PDP method is based on indexing hierarchy & homomorphic variable response method. The security of our scheme is based on trusted third party auditor and structure of zero-awareness proof , which can fulfill reliability of awareness, completeness, and properties.

Keywords
Multiple Cloud, Cooperative Provable Data Possession, Homomorphic Variable Reply, Zero awareness, Cloud Storage Security.

1. INTRODUCTION
To store data on cloud is one of the services offered by c*loud computing.* Therefore, instead of storing data on local server , they can store it on the cloud service provider's storage. Cloud storage gives the facility for users to store their data without consideration of hardware and software management, which gives advantages like capability of storing unlimited data and ability to access data anywhere[1]. Cloud computing can integrate multiple cloud services together. Thus it provides high interoperability environment. Thus this distributed cloud environment is known as *multi-Cloud.*

Cloud computing is differs from other information technology in three ways by,

A] Resources Outsourced – This includes both hardware and software. On-site file server can provide a source for file handling, data storage, and information backup.

B] Pay-as-you-can –It require a basic starting fee followed by a monthly usage charge. User need to pay charge based on cloud time consumption and additional software features.

C] On-demand facility – In cloud computing, user can pay for what they use.

In multi-cloud environment by using different interfaces clients can access resources. Web services by using virtual infrastructure management [2] is example it. Various tools and technologies are available for multiple clouds such as VMware, vSphere, and Platform VM Orchestrator[2]. These tools help cloud providers for creating a platform for data storage. But, if such an important platform is susceptible to security attacks, these attacks may introduce irrevocable losses to the clients.

The biggest issues with cloud data storage is that of data integrity verification at untrusted servers. For example, the cloud service provider (CSP) might suffers Byzantine failures. Such CSP's hide the data errors from the clients for the benefit of maintaining their reputation or for saving money and storage space the service provider might neglect to keep or deliberately delete rarely accessed data files which belong to an ordinary client. Thus, Security of the data stored over cloud is necessary for cloud service providers. Different techniques like Provable data possessions [3] (or proofs of retrievability [4]) are important for a storage provider to prove the integrity and ownership of clients' data without downloading it. This property of checking proof without downloading makes large-size files and folders to check whether these data have been tampered with or deleted without any need of downloading the data. This leads to replace traditional hash and signature functions in data storageoutsourcing.

Some recently proposed schemes like Scalable PDP [5] and Dynamic PDP [6][7] mainly focus on PDP issues at untrusted servers in a single cloud storage provider. So they are not suitable for a multi-cloud environment.

2. LITERATURE SURVEY
In this section we are presenting the different methods which are previously used for Provable Data Possesion techinique.We discuss some limitations and advantages of these systems.

Researchers have proposed two basic approaches to verify availability and integrity of outsourced data in cloud storages, called Provable Data Possession (PDP) [2] and Proofs of Retrievability (POR) [3].They have addressed the problem in distributed cloud environments of provable data possession as per following aspects: high security, transparent verification, and high performance. B.Sotomayor[1], they present OpenNebula, an open source virtual infrastructure manager that can be used to deploy virtualized services on both a local pool of resources and on external IaaS cloud. Ateniese et al. [2] proposed the PDP model for ensuring possession of untrusted storages. They also proposed a publicly verifiable version, which allows anyone, to challenge the server for data possession. This property greatly extended application areas of PDP protocol due to the separation of data owners and the users. However, these schemes are insecure against replay attacks in dynamic scenarios because of the dependencies on the index of blocks. Moreover, they do not fit for multi-cloud storage due to the loss of homomorphism property in the verification process.

In order to support dynamic data operations, Ateniese et al. developed a dynamic PDP solution called Scalable PDP [4]. They proposed a lightweight PDP scheme based on cryptographic hash function and symmetric key encryption, but the servers can deceive the owners by using previous metadata or responses due to the lack of randomness in the challenges. The numbers of updates and challenges are limited and fixed in advance and users cannot perform block insertions anywhere. Based on this work, Erway et al. [5] introduced two Dynamic PDP schemes with a hash function

tree to realize O(log1) communication and computational costs for a -block file. The basic scheme, called DPDP-I, retains the drawback of Scalable PDP, and in the 'blockless' scheme, called DPDP-II.

Furthermore, these schemes are also not effective for a multi-cloud environment because verification path of the challenge block cannot be stored completely in a cloud. Juels and Kaliski [3] presented a POR scheme, which relies largely on preprocessing steps that the client conducts before sending a file to a CSP. Unfortunately, these operations prevent any efficient ex-tension for updating data.

3. VERIFICATION FRAMEWORK OVERVIEW

In this architecture, we consider the existence of multiple CSPs to cooperatively store and maintain the clients' data. Moreover, a PDP is used to verify the integrity and availability of their stored data in all CSPs.

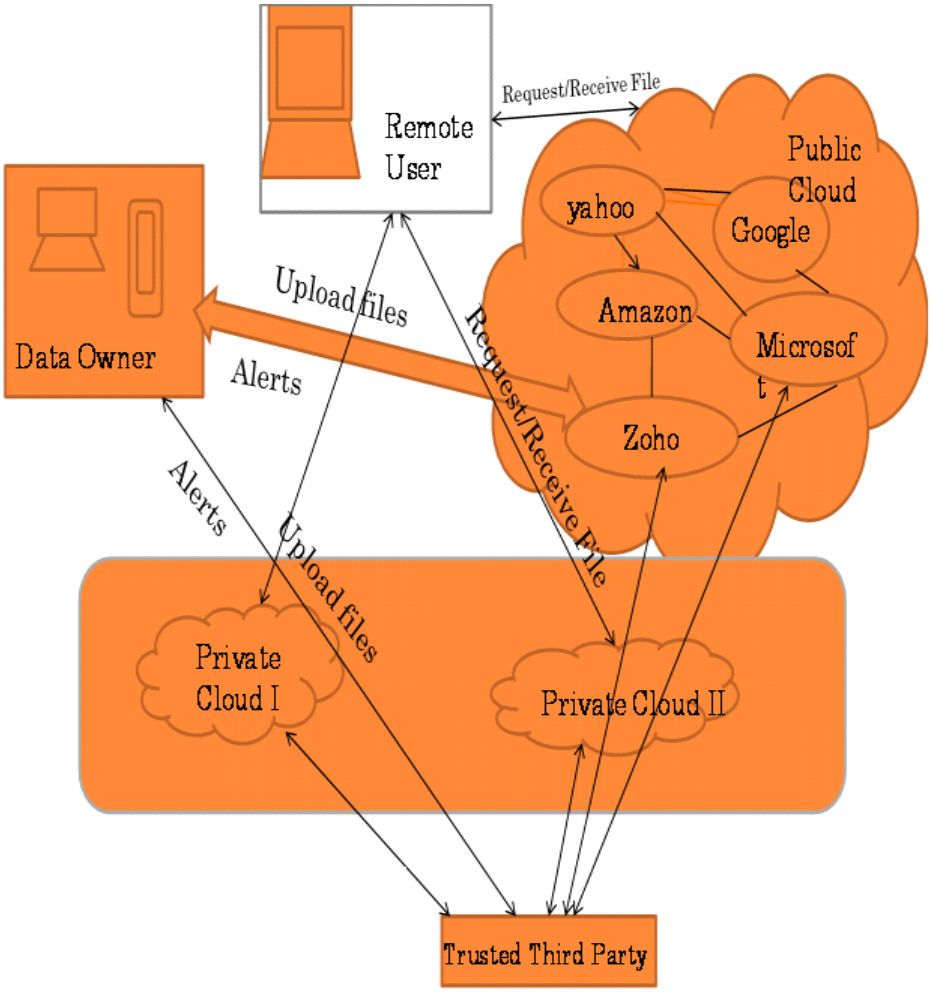

Fig. 1 System Architecture

We assume the TTP is reliable and independent through the following functions to setup and maintain the CPDP cryptosystem; to generate and store data owner's public key; and to store the public parameters used to execute the veriffcation protocol in the CPDP scheme. The TTP is not directly involved in the CPDP scheme in order to reduce the complexity of cryptosystem.

The verification procedure is described as follows: Firstly, a client (data owner) uses the secret key to pre-process a file which consists of a collection of n blocks, generates a set of public verification information that is stored in TTP, transmits the file and some verification tags to CSPs, and may delete its local copy.

Then, by using a verification protocol, the clients can issue a challenge for one CSP to check the integrity and availability of outsourced data with respect to public information stored in TTP. Then neither assume that CSP is trust to guarantee the security of the stored data, nor assume that data owner has the ability to collect the evidence of the CSP's fault after errors have been found. To achieve this goal, a TTP server is constructed as a core trust base on the cloud for the sake of security.

4. COOPERTIVE PROVABLE DATA POSSESSION SHCEME

A PDP is a collection of two algorithms (Key Gen, Tag Gen) and interactive proof system Proof.

Key Gen: It takes a security parameter as an input and returns a secret key as output.

Tag Gen: It takes a secret key, file and set of cloud storage providers as input and returns triples.

Proof: It is a protocol of proof of data possession between the CSP's and veriffer.

Let H = Hk be a family of hash functions where k : {0 1}1 index by k ε K.

This algorithm has a benefit in breaking the collision resistance of H.Collision-Resistance H: In this a hash family H(t, ε) collision resistant if no t-Time adversary has advantage atleast ε in breaking collision of H.First the KeyGen algorithm is run in this scheme to obtain the public or the private keyfor users. Then TagGen is generated by the clients for the outsourced data.

5. HOMOMORPHIC VERIFIABLE RESPONSE FOR PDP

A homomorphism is a map f : P Q between two groups such that f (g1+ g2) = f(g1) x f(g2) for all g1,g2 ε P, where + denotes the operation in P and x denotes the operation in Q. Homomorphic verifiable response is the key technique of CPDP because it not only reduces the communication bandwidth, but also conceals the location of outsourced data in the distributed cloud storage environment.

6. HASH INDEX HIERARCHY FOR CPDP

Three layers are used to illustrate the relationships among the blocks for stored resources . They are as follows:

1.Express Layer: it shows representation of stored resources.

2.Service Layer : it offers and manages cloud storage and services and

3.Storage Layer : realizes data storage on physical devices

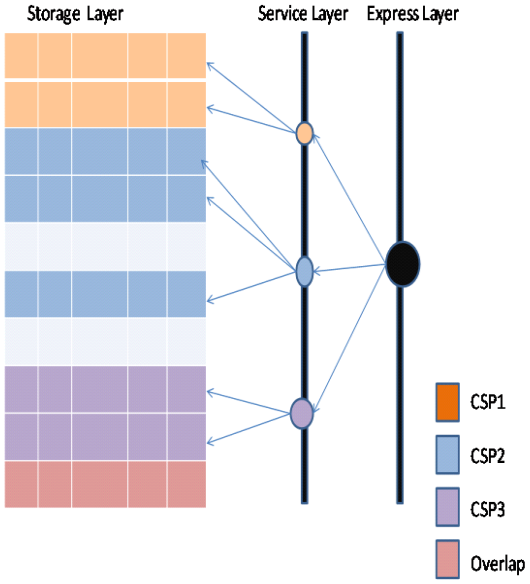

Fig. 2 Hash Index Hierarchy

7. Mathematical Model

This proposed work uses below function as follows.

1. KeyGen()

2. TagBlock()

3. GenProof()
4. CheckProof()

In this model, Symbol C is denote set component which used in this framework.

C={pk,sk,m,Tm,F,chal,Σ,V}

Where,

pk = public key,

sk = secret key,

m = file block,

Tm = metadata,

F = collection of blocks ,

Chal = challenge,

Σ = verification metadata corresponding to blocks in F,

V= proof of possession

A PDP system can be constructed from a PDP scheme in two Phases, Setup and Challenge

A . Setup:-

Pk and sk runs by client i.e KeyGen

TagBlok(): Input={pk,F, Σ=(Tm1,……..,Tmn)}

Output={F,Σ}

B.Challenge:-

Client sends challenge chal to Server S.

Genproof():-Input={pk,F,chal,Σ)

Output={F,Σ}

CheckProof():-Input={pk,sk,chal,V}

Output={success,failure}

Following polynomial-time algorithms will be used :-

◦ KeyGen(1k) → (pk,sk)

◦PrepareUpdate(sk,pk,F,info,Mc) →{e(F),e(info),e(M)}

◦PerformUpdate(pk,Fi−1,Mi−1,e(F),e(info),e(M)) →{Fi,Mi,M′c,PM′c}

◦ VerifyUpdate(sk,pk,F,info,Mc,M′c,PM′c) →{accept,reject}

◦ Challenge(sk,pk,Mc) → {c}

◦ Prove(pk,Fi,Mi,c) → {P}

◦ Verify(sk,pk,Mc,c,P) → {accept,reject}

8. SECURITY ANALYSIS:

For security purpose, cooperative scheme satisfies following properties:

1) Collision resistant indexing: The indexing hierarchy in CPDP scheme is collision resistant. If the client generates files with the same file name and tries to store in multicloud, collision because of name doesn't occur there.

2) Public verification property: This Public verification property allows client as well as anyone other than client (data owner) to challenge the cloud server for data integrity and data ownership without the need for any secret information.

3) Zero-awareness property: Privacy of the data blocks stored in multi-cloud and signature tags can be preserved by using this verification property.

4) Knowledge reliability verification: It is not possible to fool the verifier easily to accept false statements. These structures can also oppose the tag forgery attacks, which help to avoid cheating the CSPs' owner. This property is responsible for avoiding tampering of the data or tag forgery, when collisions tried.

9. MODULE INFORMATION

Module1. Login and Registration

In this we will develop the Login and Registration GUI for Entities included in Project.

Module2. Cloud Customer

The Customer or User of the Cloud is one who has a large amount of data to be stored in multiple clouds and have the permissions to use and access stored data. Before uploading process, User's Data is converted into data blocks. That data blocks are uploaded over multiple clouds in uploading process. The TTP outlooks the data blocks Uploaded in multi cloud. The user can also update the data uploaded over multiple clouds. If the client wishes to download their files, the data from multi cloud is integrated sequentially and downloaded.

Module3. Trusted Third Party

Trusted Third Party (TTP) who is trusted to store verification parameters and offer public query services for these parameters. In this system the Trusted Third Party, outlook the user data blocks and uploaded to the distributed cloud. In distributed environment of cloud each cloud has user data blocks. If anybody tries to change the data stored over cloud Trusted Third Party gets alert of it that is again sent to client.

Module4. Multi cloud storage

In this system the each cloud admin will be having data blocks stored over their cloud. Cloud computing has the ability to integrate multiple cloud services together to provide high interoperability environment as it is established based on open architectures and interfaces. Such distributed cloud environment where multiple clouds are working cooperatively is known as multi-Cloud. In this section, user uploads the data into multi cloud.

10. RESULT

In our PDP scheme, the client's communication overhead is not changed and the interaction among CSPs needs c−1 times constant-size communication overheads, where c is the number of CSPs in multi clouds. Thus, the amount of communication overheads is not increased. Further, we evaluated the performance of our PDP scheme in terms of computational overhead. For comparison, our experiments will executed as follows: a fixed-size file is used to generate the tags and prove data possession under the different number of sectors s. Then, the computational overheads of tag generation are created. The results shows that the overheads are reduced as the values of s are increased.

11. CONCLUSION AND FUTURE WORK

From research, we have presented an efficient method for security of data outsourced over multi-cloud. This research, efficient method of PDP scheme is constructed for distributed cloud storage. This scheme provided all security properties required by zero knowledge interactive proof system. . Based on indexing, we have planned a cooperative scheme to support dynamic scalability using multiple storage servers. There are multiple cloud service providers for multiple clouds as we are using multiple clouds.

Central Cloud Service Provider is used for minimizing the complexity as we want to store data block in each cloud, the request has to go from each Cloud Service Provider. Thus, Cloud Service Providers manages requests. During uploading and downloading User has to answer the Security Question. Security Questions and Answers are submitted by user during the registration phase. So during Uploading/Downloading operation If user is normal then he can answer that security questions if he/she is intruder then he/she cannot answer that questions. Thus, using this we can provide more Security. Also, we can use encryption algorithm [9] to provide the Security to uploaded data.

12. REFERENCES

[1] Yan Zhu, Hongxin Hu, Gail-Joon Ahn, "Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage" *IEEE*, Mengyang Yu, Dec-2012

[2] B. Sotomayor, R. S. Montero, I. M. Llorente, and I. T. Foster, "Virtual infrastructure management in private and hybrid Clouds," *IEEE Internet Computing*, vol. 13, no. 5, pp. 14–22, 2009.

[3] G. Ateniese, R. C. Burns, R. Curtmola, J. Herring, L. Kissner, Z. N. J. Peterson, and D. X. Song, "Provable data possession at untrusted stores," in *ACM Conference on Computer and Communications Security*, P. Ning, S. D. C. di Vimercati, and P. F. Syverson, Eds. ACM, 2007, pp. 598–609.

[4] A.Juels and B. S. K. Jr., "Pors: proofs of retrievability for Large files," in *ACM Conference on Computer and Communications Security*, P. Ning, S. D. C. di Vimercati, and P. F. Syverson, Eds. ACM, 2007, pp. 584–597.

[5] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession," in *Proceedings of the 4th international conference on Security and privacy in Communication networks, Secure Comm*, 2008, pp. 1–10.

[6] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S. S. Yau, "Dynamic audit services for integrity verification of outsourced Storages in clouds," in *SAC*, W. C. Chu, W. E. Wong, M. J. Palakal, and C.-C. Hung, Eds. ACM, 2011, pp. 1550–1557.

[7] C. C. Erway, A. K¨upc¸ ¨u, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in *ACM Conference on Computer and Communications Security*, E. Al-Shaer, S. Jha, and A. D. Keromytis, Eds. ACM, 2009, pp. 213–222.

[8] L. Fortnow, J. Rompel, and M. Sipser, "On the power of multiprover Interactive protocols," in *Theoretical Computer Science*, 1988, pp. 156–161.

[9] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Advances in Cryptology (CRYPTO'2001)*, vol. 2139 of LNCS, 2001, pp. 213–229.