An Optimal Cooperative Provable Data Possession Scheme for Distributed Cloud Storage

Shaik Faizulla[1], G. Sreedevi[2]
[1]*M. Tech, Nimra College of Engineering & Technology, Vijayawada, A.P., India.*
[2]*Assoc. Professor, Dept.of CSE, Nimra College of Engineering & Technology, Vijayawada, A.P., India.*

ABSTRACT: *In recent years, cloud storage services has become a faster growth and makes large profits by providing the services to the customer for their needs on the internet and data sharing. Cloud environment provides the services at low cost, scalable position independent platform for the storage of client's data. Though it provides good services, cloud environment is based on the open architecture and with interfaces. It has capabilities to incorporate with multiple internal or external services together to provide the ability to exchange data's and use information so that we call such a distributed cloud environments as a multi- cloud or hybrid cloud. Provable data possession (PDP) is a technique for ensuring the integrity of data in cloud environment. Provable data possession is such a probabilistic proof technique for a storage provider to prove the integrity and ownership of the clients' data without downloading data. We present an optimal cooperative PDP (CPDP) scheme based on homomorphic verifiable response and hash index hierarchy.*

Keywords: *Cloud storage, Hash index, Hybrid cloud, PDP.*

I. INTRODUCTION

Cloud computing environment is made on the basis of exposed designs and boundaries, it has the proficiency to integrate numerous interior or exterior cloud computing services together to offer great interoperability and such a dispersed cloud environment is known as a multi-Cloud [1]. By means of virtual infrastructure management a distributed permits clients to effortlessly access their resources distantly over boundaries. Cloud storage package has developed into a quicker profit progression point by providing low-cost, accessible, position self-determining stage for the customers' records. It would fetch irreversible sufferers to the clients if a hybrid cloud storage platform is susceptible to safety attacks. The secured data in an enterprise may be illegitimately opened over a remote edge delivered by a distributed cloud and documentations may be mislaid with when they are deposited into an undefined storage group external to the enterprise [2] [3].

Several implements and skills were present for hybrid cloud which aid cloud providers to build a distributed cloud storage platform for handling customers' statistics [4]. The proof checking without moving makes it particularly significant for very large- size files and folders to check whether these data have been interfered with or deleted without transferring the latest version of data. As a consequence, it is able to substitute outdated hash and signature functions in the storage outsourcing. It is necessary for cloud computing providers to offer safety practices for handling their storage facilities. Provable data possession (PDP) or proofs of retrievability is a probabilistic proof system for a storage provider to demonstrate the integrity and ownership of clients' data without transferring data [5].

II. RELATED WORK

In [6], the authors proposed the first proof-of-retrievability schemes with full proofs of security. Their first scheme, built from BLS signatures and secure in the random oracle model has the shortest query as well as response of any proof-of-retrievability with public verifiability. Their second scheme, which builds elegantly on the pseudorandom functions (PRFs) and is secure in the standard model, has shortest response of any proof-of-retrievability scheme with private verifiability but a longer query. In [4], the authors constructed a highly efficient and provably secure PDP technique based entirely on symmetric key cryptography. Also, our data possession technique allows outsourcing of dynamic data, it efficiently supports operations, such as block modification, append and deletion. In [2], the authors introduced a model for provable data possession (PDP) that 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 data possession by sampling random sets of blocks from the server, which significantly reduces I/O costs.

In [3], the authors constructed a dynamic provable data possession (DPDP), which extends the PDP model to support provable updates on the stored data. DPDP solution is based on a new variant of the authenticated dictionaries which use rank information to organize dictionary entries. Thus we can support efficient authenticated operations on the files at block level, such as authenticated insert and delete. They proved the security of their constructions using some standard assumptions. In [7], the authors proposed an innovative authentication scheme for mobile devices. While creating the password, the user chooses a theme of snapshots in thumbnail size and the sequence of those snapshots is fixed as a password. In [8], passface is an approach proposed by the Real User Corporation in which the user is allowed to choose four images of human faces from the face database as their password. During the verification step, the user is provided with a grid of nine faces, one already chosen during the registration and eight decoy faces. The user identifies the selected face and then clicks anywhere over it.

III. PROPOSED WORK

A. Verification Framework for Multi- Cloud

In the proposed work, we consider a data storage service involving three different entities: Granted clients, who have a large amount of data to be stored in multi-clouds and have the permissions to access and manipulate the stored data; Cloud service providers (CSPs), who work together for providing data storage services and have enough storage space sand computation resources; and Trusted third parties (TTPs),who are trusted to store the verification parameters and offer the query services for these parameters. Figure 1 shows the architecture.

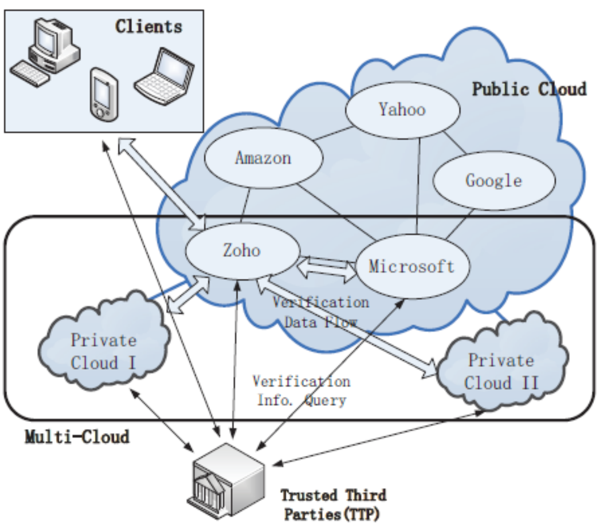

Figure 1: Verification architecture for data integrity

In this architecture, we consider the existence of many CSPs to collaboratively store and maintain the client‟ s data. Moreover, a cooperative PDP is used to verify the integrity and availability of the stored data in CSPs. The verification process is described as follows: Firstly, the client (data owner) uses the shared secret key to pre-process the file, which consists of a collection of n blocks. It generates a set of public verification information that is stored in TTP and then transmits the file and some verification tags to CSPs, and may delete its local copy. Later by using a verification protocol for cooperative PDP, the clients can issue a challenge for one CSP to check the integrity and the availability of outsourced data in terms of public verification information stored in TTP.

B. Cooperative PDP

Based on zero knowledge proof system and interactive proof system, the integrity of data stored in a multi cloud is maintained. A CPDP is a collection of two algorithms (Key Gen, Tag Gen) and the interactive proof system Proof.

- **Key Gen:** It takes a security parameter as an input and then returns a secret key as output.
- **Tag Gen:** It takes a shared secret key, file and set of cloud storage providers as input and returns triples.
- **Proof:** It is a protocol of proof of provable data possession between the CSP's and verifier.

Let H = { Hk } be a family of hash functions where Hk : $\{0,1\}^n \rightarrow \{0,1\}^k$ index, where k ε K. This algorithm has a benefit in breaking the collision resistance of hash H. In Collision-Resistance H, 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 private or the public key for users. Then TagGen is generated by the clients for the cloud outsourced data.

C. Homomorphic Verifiable Response

A homomorphism is a map function f : p → q between two groups such that f(g1 $\oplus$ g2) = f(g1) $\otimes$ f(g2) for all g1, g2 ∈ p, where $\oplus$ denotes the operation in p and $\otimes$ denotes the operation in q. This notation was used to define a Homomorphic Verifiable Tags (HVTs) as : Given two values σ_i and σ_j for two message mi and mj , anyone can combine them into a value σ' corresponding to the sum of the message mi+mj . When provable data possession is considered as a challenge response protocol, we also extend this notation to introduce the concept of a Homomorphic Verifiable Responses (HVRs), which is used to integrate multiple responses from the various CSPs in cooperative PDP scheme, as follows:

A response is called homomorphic verifiable response in data possession protocol, if given two responses θi and θj for two challenges Qi and Qj from two CSPs, there exists an efficient algorithm to combine them into a response θ corresponding to the sum of the challenges Qi uQj.

D. Fragment Structure of CPDP

We propose a fragment structure of our data possession scheme based on the above-mentioned model as shown in Figure 2, which has following characters: 1) a file is split into n × s sectors and each block (s sectors) corresponds to a tag, so that the storage of the signature tags can be reduced with the order of s; 2) the verifier can check the integrity of a file by random sampling approach, which is a matter of the utmost importance for large or huge files; and 3) this structure relies on the homomorphic properties to aggregate the data and tags into a constant size response, which minimizes network communication overheads. The above structure, considered as a common representation for some of the existing schemes, can be converted to MAC-based, ECC or RSA schemes. By using BLS signatures and random oracle model, it is easy to design a practical data possession scheme with the shortest homomorphic verifiable responses for public verifiability. This structure also creates some favorable conditions for the architecture of CSPs.

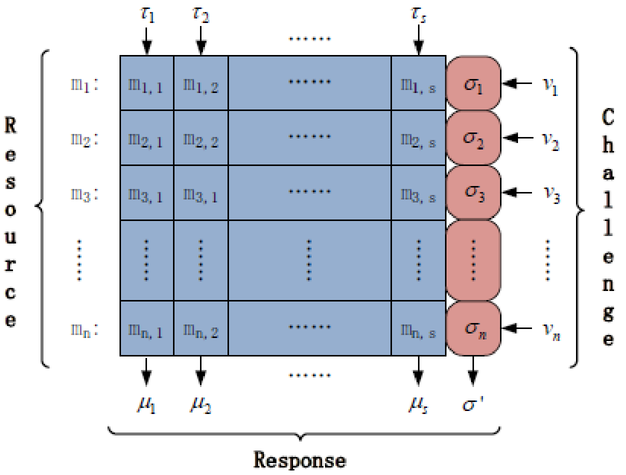

Figure 2: The fragment structure of CPDP model

E. Hash Index Hierarchy

A representative architecture for data storage in distributed is illustrated as follows: this architecture is a hierarchical structure H on 3 layers to represent the relationships among all blocks for stored resources. Three layers are as follows:

- **First-Layer (Express Layer):** It offers an abstract representation of the stored resources;
- **Second-Layer (Service Layer):** It immediately offers and manages cloud storage services;
- **Third-Layer (Storage Layer):** It practicably realizes data storage on many physical devices

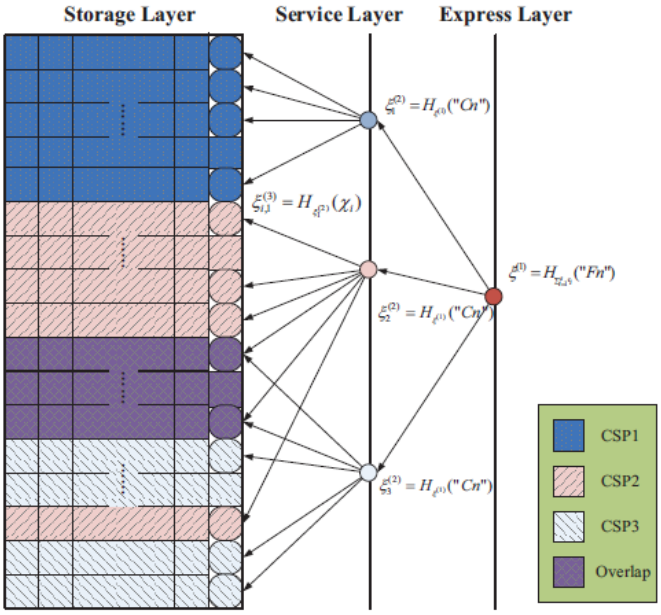
Storage Layer
Service Layer
Express Layer
$\xi_1^{(2)} = H_{\xi^{(1)}}("Cn")$
$\xi_{1,1}^{(3)} = H_{\xi_1^{(2)}}(\chi_1)$
$\xi^{(1)} = H_{\Sigma_{i=1}^{s}}("Fn")$
$\xi_2^{(2)} = H_{\xi^{(1)}}("Cn")$
$\xi_3^{(2)} = H_{\xi^{(1)}}("Cn")$
CSP1
CSP2
CSP3
Overlap

Figure 3: Index-hash hierarchy of CPDP model

Figure 3 shows index hash hierarchy of our model. This kind of architecture is a nature representation of the file storage. We make use of this simple hierarchy to organize many CSP services, which involve private clouds or public clouds, by shading the differences between these clouds. In this architecture, the resources in Express Layer are split and stored into

3 CSPs in Service Layer. Each CSP fragments and stores the assigned data into the cloud storage servers in Storage Layer. We follow the logical order of the data blocks to organize the cloud Storage Layer. Moreover, this proposed architecture could provide some special functions for data storage and management.

IV. CONCLUSION

In this paper, we proposed the scheme of optimal cooperative provable data possession and its distributed cloud storage to support the qualities of the services provides. In this work, we consider and maintains the existence of multiple cloud service providers to the cooperatively stores and it's also maintain the clients data in safe and security. We presented the scheme based on homomorphic verifiable response and hash index hierarchy. The security of our scheme is based on the multi- prover zero-knowledge proof performance optimization mechanisms for our scheme. In particularly, increasing the efficiency for our proposed scheme and minimize the system, which can satisfy completeness, knowledge soundness, and zero-knowledge properties.

REFERENCES

[1] 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.

[2] 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.

[3] C.C.Erway, A.Ku¨pc¸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.

[4] G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, “Scal- able and efficient provable data possession,” in Proceedings of the 4th international conference on Security and privacy in communication netowrks, SecureComm, 2008, pp. 1–10.

[5] H. Shacham and B. Waters, “Compact proofs of retrievabil- ity,” in ASIACRYPT, ser. Lecture Notes in Computer Science, J. Pieprzyk, Ed., vol. 5350. Springer, 2008, pp. 90–107.

[6] H. Shacham and B. Waters, “Compact Proofs of Retrievability,” Proc. 14th Int‟ l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT ‟ 08), pp. 90-107, 2008.

[7] W. Jansen, "Authenticating Mobile Device User through Image Selection," in Data Security, 2004.

[8] Real User Corporation: Passfaces. www.passfaces.com