Remote Data Integrity Checking in Multi Cloud Storage based on Identity Distributed Provable Data Possession

Ms. Maitra S Sampagavi[1], Ms. Sangeetha S[2], Ms. Devi T[3]
[1, 2,]UG students, department of computer science, Rajarajeswari college of engineering India
[3]Professor, department of computer science, Rajarajeswari college of engineering India

***Abstract*- The client cannot store a large amount of data in a local machine, because it is difficult for him to maintenance and estimation. This is why the reason client offers a cloud. When the client storing data in a cloud, the client will be having a security issues like data integrity, confidentiality etc., In some case, the clients have to store a data in multi cloud severs. At the same time, the integrity checking protocol must be efficient in order to save verifier cost and distributed storage and integrity checking are absolutely necessary. Distributed Provable Data Possession is technique for safeguard the integrity of data stored in a multi cloud server. It can make the clients verify whether their outsourced data is secure or not without downloading the whole data. From the above points, we propose a novel protocol: RDI-IDPDP (remote data integrity checking in multi cloud storage based on identity Distributed Provable Data Possession) in multi cloud storage. Based on distributed computation and Provable Data Possession, a concrete RDI-IDPDP protocol is designed. The Proposed RDI-IDPDP protocol is secure under the hardness assumption of CDH (computational Diffie-Hellman) problem, the easiness of DDH (Decisional Diffie-Hellman) problem and GDH (Gap Diffie-Hellman) groups. Our RDI-IDPDP protocol is also efficient and adoptable. Based on the client's permission, the proposed RDI-IDPDP protocol can perceive client verification, third party verification and Identity-based public key cryptography can eliminate the complicated certificate management.**

Index Terms- Distributed computing, Cloud computing, Provable data possession, Identity-based cryptography, Bilinear pairing

I. INTRODUCTION

Over the last year, the cloud computing becomes a emergent technology in computer field. It is typically defined as a type of computing that relies on sharing computing resource rather than having local server or personal devices to application. In cloud computing the word cloud is used as a metaphor for the "Untrusted network", so the cloud computing means 'Untrusted Network -based Computing'.

The different services such as server, storage and application are delivered to an organisation's computer and device's through the internet. Cloud storage service has become a faster profit growth point by providing a comparably Resource pooling and elasticity, Pricing, Quality of server, self service and on-demand services. At the same time, it avoids of capital setback on hardware, software, and personnel maintenances, etc., the client delegate the computing task to the third party. It involves a security issues in terms of affinity, incorruptibility, possibility of data and service.

Since the client data is outsourced to the cloud, the data incorruptibility is the main issue to the client because the clients do not store these data locally. In some general case, when the client stores his data on multi-cloud servers, the distributed computation storage and integrity checking are absolutely necessary. At the same time, the integrity checking protocol must be efficient in order save a client, verifier cost. Thus, based on distributed computation, provable data possession we present the corresponding concrete protocol in multi-cloud storage.

A. Motivation

We consider an EMC in the cloud computing environment. EMC has carefully selected Cloud Service Providers to help client. The EMC provide following services: protection of client data, Safety of data, framework, Applications, Desktop, Big Data, etc., In addition to above services the EMC has some public information, hybrid information.EMC store these data on different cloud servers. The different cloud service providers have different stature and charging standard. These cloud service providers need different charges according to the different security-level. Thus, EMC will select different cloud service providers to store its different data. For some sensitive or important data, it will copy these data many times and store these copies on different cloud servers. For the hybrid data, it will store them on the hybrid cloud server.

The public data will be stored in public cloud server. At last, EMC stores its whole data on the different cloud servers according to their importance and sensitivity. The profit and losses of EMC depend on selection of cloud server. Therefore in multi-cloud environment, distributed provable data possession is an important element to secure the remote data. In PKI (public key infrastructure), provable data possession protocol needs public key document which attesting ownership of an item or the

fulfilment of legal requirements distribution and management. It will suffer considerable problems since the verifier will check the certificate document when it checks the remote data correction in multi cloud. In addition to the heavy certificate document affirmation, the system also suffers from the other complicated certificate document management such as certificates generation, delivery, revocation, renewals, etc. Identity-based public key cryptography can eliminate the complicated certificate document management by predefining the public key using a unique identity of a client to increase the efficiency of protocol. Thus, it will be very meaningful to study the RDI-IDPDP.

B. Related Work

In cloud computing, the correctness and completeness of data is important problem. As the client data is outside his control. The harmful cloud server may corrupt the clients' data in order to gain more benefits. Many researchers proposed the protocol corresponding to system model and security model. In 2007, provable data possession (PDP) was proposed by G.Ateniese [1].The proposed PDP model is based on RSA which generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduce I/O costs. The client maintains a constant amount of metadata to verify the proofs and the proposed PDP support to check integrity to a large amount of data. After that, G.Ateniese et al. proposed a dynamic PDP [2] which does not support for insertion operation, in 2010 Y.ZHU proposed a Efficient Provable Data Possession for Hybrid Clouds [3] which support scalability of service, data migration and minimize the communication complexity.

The problem with proposed protocol is safety security of the data. Later 2010 Z.Hao Proposed "MR-PDP: Multiple Replica Provable Data Possession [4] which provides clients with the ability to check whether multiple replicas are really stored at the cloud storage servers. Which provide a client verification and flexibility in third party auditing, but there is problem for private verification. In 2009, Erway et al. proposed a full-dynamic PDP scheme based on the authenticated flip table [5]. The similar work has also been done by F. Sebe et al. [6]. PDP allows a verifier to verify remote data correctness without retrieving or downloading the whole data.

It is a probabilistic proof of possession by sampling random set of blocks from the server, which drastically reduces I/O costs. The verifier only maintains small metadata to perform the integrity checking. PDP is an interesting remote data integrity checking model. In 2012, Wang proposed the security model and concrete scheme of proxy PDP in public clouds [7] which is based on bilinear pairing technique, this protocol is efficient when client cannot perform the remote data possession checking.

Along this many remote data integrity checking models and protocols have been proposed [8] [9] [10]. In 2008, H. Shacham, proposed a compact proofs of retrievability, which check and retrieve remote data any time [11]. On some cases, the client may delegate the remote data integrity checking task to the third party. It results in the third party auditing in cloud computing. In 2011, Y. Zhu, proposed a Dynamic Audit Services for Outsourced Storages in Clouds [12] which provide a audit service based on fragment structure, random sampling technique. Clients with the help of TPA to verify the correctness of cloud data but it cannot support data checking without local copy. In 2009, C. Wang Proposed a Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing [13] which enable the external auditor to audit user's outsourced data in the cloud without learning the data content but it does not support for a public audit ability.

C. Contributions

In identity-based public key cryptography, it predefines public key by using a unique identity of client, this paper focuses on distributed provable data possession in multi-cloud storage. The protocol can be made efficient by eliminating the certificate document management. We propose the new remote data integrity checking model: RDI-IDPDP. Then, based on the bilinear pairings, the concrete RDI-IDPDP protocol is designed. In the random oracle model, our RDI-IDPDP protocol is provably secure. On the other hand, our protocol is more flexible besides the high efficiency. Based on the client's authorization, the proposed RDI-IDPDP protocol can realize client verification, delegated verification and TPA verification.

II. SYSTEM MODEL

In The RDI-IDPDP system model is presented in this section. An RDI-IDPDP protocol comprises five different entities which are illustrated in figure 1. We describe them as follows:

PKG: Entity, trusted by the clients and the PCSs, that generates the public parameters Parma's, the master public key mpk, the master secret key msk and the private key of the Client and also receives an outputs.

CLIENT: Entity which has to store a huge amount of data on multi cloud for maintenance and computation, it can either individual consumer or corporation. **e.g.**, the departments of the company in the motivated scenario.

CS (Cloud Server): Entity, which is controlled and maintained by the cloud service provider that has significant storage space and computational resources to maintain the clients' data.

In the cloud paradigm, by putting the large data files on the remote cloud servers, the clients can be relieved of the burden of storage and estimation. As the client's data is out of his control, they no longer possessing their data locally, it is of critical importance for them to ensure that their data are being correctly stored and preserved. That is, clients should be equipped with certain security means so that they can periodically verify the correctness of the remote data even without the existence of local copies.

COMBINER: An entity, which receives the storage request and distributes the block-tag pairs to the corresponding cloud servers. When receiving the challenge, it splits the challenge and distributes them to the different cloud servers. When receiving the responses from

the cloud servers, it combines them and sends the combined response to the verifier.

VERIFIER: Is run by the PCS in order to generate a proof of data possession. It takes as inputs the public Parameter, the client's identity ID, an ordered collection of blocks, a challenge and an ordered collection of the verification tag corresponding to the blocks in. It returns a proof of data yield for the blocks in that are determined by the challenge.

The verifier should not be required to keep an entire copy of the file(s) to be checked. It would be impractical for a verifier to replicate the content of all prover to be verified.

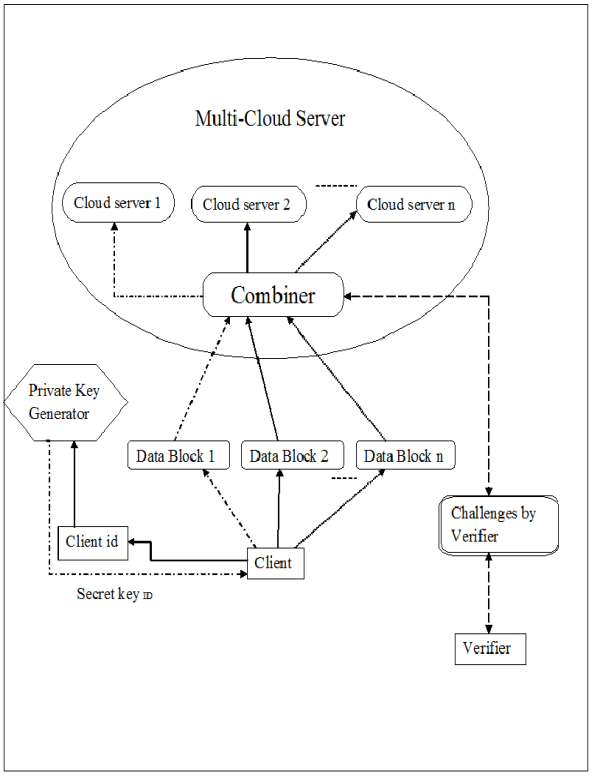

Figure1. The System Model of RDI-IDPDP

An ID-DPDP protocol is a collection of three algorithms (Setup, Extract,Tag Gen) and an interactive proof system (Proof). They are described in detail below.

1) Setup (1k): Input the security parameter k, it outputs the system public parameters params, the master public key mpk and the master secret key msk.

2) Extract (1k, params, mpk, msk,ID): Input the public parameters params, the master public key mpk, the master secret key msk, and the identity ID of a client, it outputs the private key skID that corresponds to the client with the identity ID.

3) TagGen (skID, Fi,P): Input the private key skID, the block Fi and a set of CS P = {CSj}, it outputs the tuple {φi,(Fi,Ti)}, where φi denotes the i-th record of metadata, (Fi,Ti) denotes the i-th block-tag pair. Denote all the metadata {φi} as φ.

4) Proof (P, C (Combiner), V (Verifier)): is a protocol among P, C and V . At the end of the interactive protocol, V outputs a bit {0|1} denoting false or true.

III.CONCLUSION

In this paper RDI-IDPDP illustrate system model, the proposed RDI-IDPDP protocol can realize TPA verification, client verification and delegated verification based on client authorization. RDI-IDPDP protocol is provably secure under the harness assumption of standard CDH problem. Besides from this it also eliminates certificate management using Identity based crypto-graphy in which the overall process is performed in multi-cloud storage. RDI-IDPDP protocol has also flexibility and high efficiency

IV.REFERENCES

[1] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, D. Song, "Provable Data Possession at Untrusted Stores", CCS'07, pp. 598-609, 2007.

[2] G. Ateniese, R. DiPietro, L. V. Mancini, G. Tsudik, "Scalable and Efficient Provable Data Possession", SecureComm.

[3] Y. Zhu, H. Wang, Z. Hu, G. J. Ahn, H. Hu, S. S. Yau, "Efficient Provable Data Possession for Hybrid Clouds", CCS'10, pp. 756-758, 2010.

[4]R. Curtmola, O.Khan, R.Burns, G.Ateniese,"MR-PDP: MultipleReplica Provable Data Possession", ICDCS'08.

[5] C.C.Erway, A. Kupcu, C. Papamanthou, R. Tamassia, "Dynamic Provable Data Possession", CCS'09, pp. 213-222, 2009.

[6] F. Sebe´, J. Domingo-Ferrer, A. Mart´ınez-Balleste´, Y. Deswarte, J. Quisquater, "Efficient Remote Data Integrity checking in Critical Information Infrastructures", IEEE Transactions on Knowledge and Data Engineering, 20(8),pp.1-6,2008.

[7] H.Q. Wang, "Proxy Provable Data Possession in Public Clouds," IEEE Transactions on Services Computing,2012. http://doi.ieeecomputersociety.org/10.1109/TSC.2012.35

[8] Y. Zhu, H. Hu, G.J. Ahn, M. Yu, "Cooperative Provable Data Possession for Integrity Verification in Multicloud Storage", IEEE Transactions on Parallel and Distributed Systems, 23(12), pp. 2231-2244, 2012.

[9] A. F. Barsoum, M. A. Hasan, "Provable Possession and Replication of Data over Cloud Servers", CACR, University of Waterloo , Report2010/32,2010.

[10] Z. Hao, N. Yu, "A Multiple-Replica Remote Data Possession Checking Protocol with Public Verifiability", 2010 Second International Symposium on Data, Privacy, and E-Commerce, pp. 84-89, 2010.

[11] H. Shacham, B. Waters, "Compact Proofs of Retrievability", ASIACRYPT 2008, LNCS 5350, pp. 90-107, 2008.

[12] Y. Zhu, G.J. Ahn, H. Hu, S.S. Yau, H.G. An, S. Chen, "Dynamic Audit Services for Outsourced Storages in Clouds," IEEE Transactions on Services Computing.

[13] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing", INFOCOM 2010, IEEE, March 2010.