# Identity-Based Public Multi-Replica Provable Data Possession

**Saurabh Pandey[1], Remya Sivan[2]**
[1]Assistant Professor
[1, 2] Atria Institute of Technology, Visvesvaraya Technological University, Bangalore

***Abstract-*** *Cloud storage has been gaining tremendous popularity, which provides facilitative data storage and sharing services for distributed clients. To maximize the availability and reliability, some customers may store multiple replicas of critical data on cloud servers. However, cloud servers may collude to make it look like they are storing multiple copies of data, whereas in fact they only store a single copy. Currently, several multi-replica provable data possession schemes have been proposed to provide verifications to ensure that all the outsourced copies are actually stored and maintained intact. For these schemes with third-party verifications, correctly choosing public keys of data owners relies on the public key infrastructure (PKI),which is complicated and resource consuming. In this paper, we propose a novel identity-based public multi replica provable data possession scheme (IDPMR-PDP) to provide third-party verification of outsourced data with multiple replicas without PKI. We also introduce a formal security model of identity-based public multi-replica PDP schemes and prove that the IDPMR-PDP is secure against malicious cloud servers and privacy-preserving against curious verifiers under this model. Meanwhile, our analyses and simulation results demonstrate that the IDPMR-PDP realizes efficient integrity verification.*

***Keywords***- Cloud storage, identity-based cryptography, multi-replica, provable data possession.

## I. INTRODUCTION

With the rising popularity of cloud computing and its ever growing versatility, it is no surprise that an increasing number of individuals and organizations have integrated their data and services into the clouds. They use the clouds in a variety of service models (with acronyms such as SaaS, PaaS, and IaaS) and deployment models (private, public, hybrid, and community). Cloud storage saves enterprises a tremendous amount of money in IT investments, such as purchasing, management and maintenance of hardware. Mean while, with benefits like higher exibility, automatic software updates, increased collaboration and the free dom to work from anywhere, cloud storage has been gaining its prevalence in our daily life. These high demands have stimulated the cloud service providers to offer affordable elastic and remotely-accessible cloud storages, which thrives as a prot growth point in cloud computing. However, cloud storage is a double-edged sword. It brings great convenience, whereas some inherent security risks. When individuals and organizations outsource their data to the cloud storage, they do not possess the data locally anymore and lose their ability to have physical access to the servers hosting their data. These third-parties holding their sensitive business data may not be working in their best interest and may not be mandatory to report data errors. To be worse, they may conceal data loss incidents to maintain their reputation, or discard rarely accessed data to save storage costs, etc. In 2016, the cloud security alliance (CSA) listed the ``Treacherous 12", i.e., the top 12 cloud computing threats which individuals and organizations face [2]. This investigation ranked ``data breaches" and ``data loss" 1st and 8th respectively out of 12 top threats in cloud computing.

## II. RELATED WORK

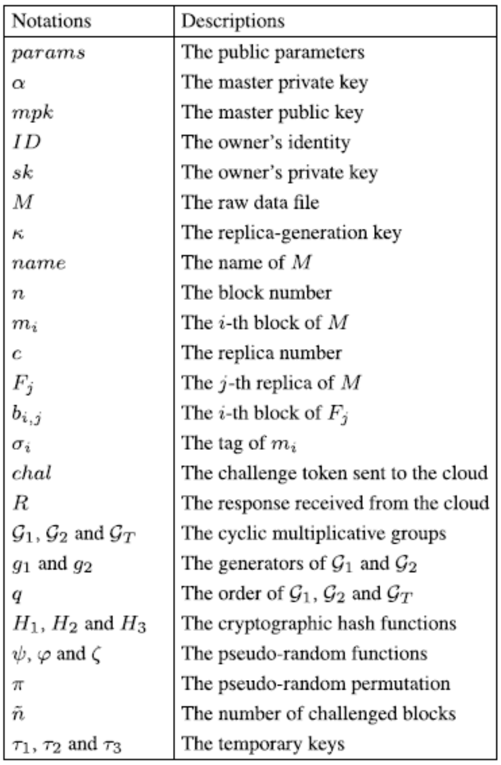
For the data owners, to confirm that their data is beings to red and maintained intact, it is necessary to develop efficient integrity verification techniques to strengthen their confidence in cloud storage. In 2007, Ateniese et al. proposed a probabilistically accurate data integrity verification method namely provable data possession, i.e., PDP, which enables data owners to check the integrity of their outsourced data remotely without downloading the entire files.

Among these works, allow the data owners with limited computation and communication power to delegate the remote integrity checking tasks to third-party verifiers. They also achieved privacy-preserving verification against the semi-trusted third-party verifiers who may be curious about the owners' data. However, to confirm if the data is uploaded by a certain owner, the aforementioned schemes rely on the public key certificates issued by Public Key Infrastructure (PKI), which ensures authenticity of public keys whereas introduces some other problems. On the one hand, PKI is complicated and the managements of certificates, such as delivery, renewal and revocation, are resource consuming. On the other hand, the security of PKI itself maybe vulnerable.

To eliminate the burden involved by PKI, some PDP schemes, like resort to the identity-based cryptosystem, in which the public keys of data owners are simply their identities, like names and e-mall addresses. In these schemes, trusted PKGs (Private Key Generators) generate the private keys for all the data owners corresponding to their identities. As a result, data owners' public keys can be self-authenticated without verifying their certificates. Among these schemes, Wang proposed the first ID-based PDP scheme in multi-cloud storage. Although it is vulnerable for some security fiaws, it is later _xed by Peng *et al.* Wang *et al.* proposed an ID-based PDP scheme allows the data owners to delegate the data uploading and integrity checking to their proxies; Yu *et al.* proposed an ID-based PDP scheme with perfect data privacy-preserving which is proven secure against malicious servers and curious verifiers.

## III. SYSTEM MODEL AND SECURITY MODEL

IDPMR-PDP involves four entities: the private key generator(PKG), the data owner (owner), the public verifier (verifier) and the cloud server (cloud). The PKG outputs the corresponding private key when receiving the identity of the owner. The owner creates the data and stores it in the cloud. The verifier is a third party to provide verification service for the owner. The cloud stores the owner's data and provides data access to the owner and, maybe, the other data users. Furthermore, we assume that the owner possesses a largeaw data le, splits the file into smaller blocks and stores the blocks with multiple replicas in the cloud.



| Notations | Descriptions |
|---|---|
| $params$ | The public parameters |
| $\alpha$ | The master private key |
| $mpk$ | The master public key |
| $ID$ | The owner's identity |
| $sk$ | The owner's private key |
| $M$ | The raw data file |
| $\kappa$ | The replica-generation key |
| $name$ | The name of $M$ |
| $n$ | The block number |
| $m_i$ | The $i$-th block of $M$ |
| $c$ | The replica number |
| $F_j$ | The $j$-th replica of $M$ |
| $b_{i,j}$ | The $i$-th block of $F_j$ |
| $\sigma_i$ | The tag of $m_i$ |
| $chal$ | The challenge token sent to the cloud |
| $R$ | The response received from the cloud |
| $\mathcal{G}_1$, $\mathcal{G}_2$ and $\mathcal{G}_T$ | The cyclic multiplicative groups |
| $g_1$ and $g_2$ | The generators of $\mathcal{G}_1$ and $\mathcal{G}_2$ |
| $q$ | The order of $\mathcal{G}_1$, $\mathcal{G}_2$ and $\mathcal{G}_T$ |
| $H_1$, $H_2$ and $H_3$ | The cryptographic hash functions |
| $\psi$, $\varphi$ and $\zeta$ | The pseudo-random functions |
| $\pi$ | The pseudo-random permutation |
| $\tilde{n}$ | The number of challenged blocks |
| $\tau_1$, $\tau_2$ and $\tau_3$ | The temporary keys |

*Definition 1 (Identity-Based Public Multi-Replica Provable Data Possession):* IDPMR-PDP is comprised of six phases:

1) Setup(1*k* ) ! (*params*; alpha _; *mpk*). The phase is run by the PKG. Identical to an ID-based signature scheme,it takes as input a security parameters *k*, outputs some public parameters *params*, a master private key _ and the corresponding master public key *mpk*. The PKG publishes *params* and *mpk*, keeps _ secret.

2) Extract(*params*; *ID*; alpha) ! *sk*. The phase is run by the PKG. *ID* denotes the owners's identity. Identical to an ID-based signature scheme, it takes as input *params,ID* and _, outputs a corresponding private key *sk* and forwards *sk* to the owner.

3) ReplicaGen (*params*;*M*; k; *name*; *c*) ! f*bi*;*j*g. The phase is run by the owner. *M* denotes a raw data file with *n* blocks, i.e., *M* D (*m*1; : : : ;*mn*) with *mi* denotes the *i*-th block, _ denotes the replica-generation key,*name* denotes the name of *M*, *c* denotes the replica number. It takes as input *params*, *M*, _, *name* and *c*,outputs a set of replica _les f*F*1; : : : ; *Fc*g of *M* with sets of replica blocks, i.e., f*Fj*g D f(*b*1;*j*; : : : ; *bn*;*j*) j *j* D 1;: : : ; *c*g. The owner forwards f*bi*;*j*g to the cloud.

4) TagGen(*params*; *ID*; *sk*; *mpk*;*M*; *name*) ! f_*i*g. The phase is run by the owner. It takes as input params, ID, sk, mpk, M D (m1; : : : ;mn) and name,outputs a corresponding set of tags

f1; :::; ng. Be different to most of the other multi-replica PDP schemes, they are generated from the raw data blocks fmig, not from the replica blocks fbi;jg. The owner forwards fig to the cloud.

5) Challenge(params; ID; mpk; name) ! chal. The phase is run by the verier. It takes as input params, ID, mpk and name, outputs a challenge token chal. The verier forwards chal to the cloud.

6) GenProof (params; fbi;jg; fig; name; c; chal) ! R. The phase is run by the cloud. It takes as input params,fbi;jg, fig, name, c and chal, outputs a response R. The cloud forwards R to the verifier.

7) Check Proof (params; ID; mpk; name; c; chal; R) !f0; 1g. The phase is run by the verier. It takes input params, ID, mpk, name, c, chal and R, outputs1 (valid) or 0 (invalid).

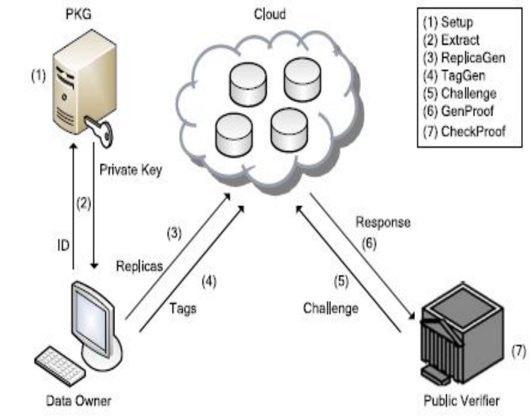


**FIGURE 1.** Architecture of IDPMR-PDP.

## REFERENCES

[1] M. K. Srinavasin, K. Sarukesi, P. Rodrigues, M. S. Manoj, and P. Revathy,``State-of-the-art cloud computing security taxonomies: A classication ofsecurity challenges in the present cloud computing environment,'' in Proc.ICACCI, 2012, pp. 470476.

[2] Cloud Security Alliance. The Treacherous 12: Cloud ComputingTop Threats in 2016. Accessed: Nov. 20, 2017.

[3] G. Ateniese et al., ``Provable data possession at untrusted stores,'' in Proc. CCS, 2007, pp. 598609.

[4] F. Sebé, J. Domingo-Ferrer, A. Martínez-Ballesté, Y. Deswarte, andJ.-J. Quisquater, ``Efcient remote data possession checking in criticalinformation infrastructures,'' IEEE Trans. Knowl. Data Eng., vol. 20, no. 8, pp. 10341038, Aug. 2008.

[4] G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, ``Scalable and efcient provable data possession,'' in Proc. SecureComm, 2008,pp. 110.

[5] C. Erway, A. Küpçü, C. Papamanthou, and R. Tamassia, ``Dynamic provable data possession,'' in Proc. CCS, 2009, pp. 213222.

[6] Y. Zhu, H.Wang, Z. Hu, G. Ahn, H. Hu, and S. S. Yau, ``Efcient provable data possession for hybrid clouds,'' in Proc. CCS, 2010, pp. 756758.

[7] Y. Zhu, H. Hu, G. J. Ahn, and M. Yu, ``Cooperative provable data possessionfor integrity verication in multicloud storage,'' IEEE Trans. ParallelDistrib. Syst., vol. 23, no. 12, pp. 22312244, Dec. 2012.

[8] K. Yang and X. Jia, ``An efcient and secure dynamic auditing protocol for data storage in cloud computing,'' IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 9, pp. 17171726, Sep. 2013.

[9] H.Wang, ``Proxy provable data possession in public clouds,'' IEEE Trans.Services Comput., vol. 6, no. 4, pp. 551559, Oct./Dec. 2013.

[10] Y. Yu, J. Ni, M. Au, H. Liu, H. Wang, and C. Xu, ``Improved security ofa dynamic remote data possession checking protocol for cloud storage,'' Expert Syst. Appl., vol. 41, no. 17, pp. 77897796, 2014.

[11] Y. Yu, Y. Zhang, J. Ni, M. H. Au, L. Chen, and H. Liu, ``Remote datapossession checking with enhanced security for cloud storage,'' FutureGenerat. Comput. Syst., vol. 52, pp. 7785, Nov. 2015.

[12] Q.Wang, C.Wang, K. Ren,W. Lou, and J. Li, ``Enabling public auditabilityand data dynamics for storage security in cloud computing,'' IEEE Trans.Parallel Distrib. Syst., vol. 22, no. 5, pp. 847859, May 2011.

[13] C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, ``Toward secure and dependable storage services in cloud computing,'' IEEE Trans. ServicesComput., vol. 5, no. 2, pp. 220232, Apr./Jun. 2012.

[14] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, ``Privacypreservingpublic auditing for secure cloud storage,'' IEEE Trans. Comput.,vol. 62, no. 2, pp. 362375, Feb. 2013.

[15] Y. Zhu, G.-J. Ahn, H. Hu, S. S. Yau, H. G. An, and C.-J. Hu, ``Dynamicaudit services for outsourced storages in clouds,'' IEEE Trans. ServicesComput., vol. 6, no. 2, pp. 227238, Apr./Jun. 2013.