International Journal of Research

International Journal of Research (IJR)

e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 10, October 2015

Available at http://internationaljournalofresearch.org

Identity-Based Integrity Verification using PDP in Multi Cloud Storage

V.Rakesh#1& P.Srinivasulu #2

#V.RAKESH of M.Tech (CSE) and Department of Computer Science Engineering,
#P.SRINIVASULUAssoc.Prof,and HOD Department of Computer Science and Engineering, AP

Abstract:

Cloud computing has become an important thing in computer field. Cloud computing takes information processing as a service, such as storage and computing. Data integrity is important thing in cloud storage. In certain situations, clients should store their data such as image or text in multi cloud. When the client stores his/her data on multi cloud servers, the distributed storage and integrity checking is very important. Here we propose an Identity Based Distributed Provable Data Possession (ID-DPDP) protocol for multi-cloud storage. Remote data integrity checking is important in cloud storage. It can make the clients verify whether their data is kept as it is without downloading the entire data. In some application scenarios, the clients have to store their data on multi-cloud servers. At the same time, the integrity checking protocol must be efficient in order to save the verifier's cost.

Keywords: Multi-cloud storage; Cooperative Provable Data Possession; Zero Knowledge Property; Hash Index Hierarchy; Homomorphic Verifiable Response

I.INTRODUCTION :

A protocol (ID-DPDP-Identity-based distributed provable data possession) is proposed to store data in multi cloud .ID-DPDP protocol eliminate the certificate management. In this system, the client's data is distributed to multi cloud servers based on er Science and Mobile Computing, type of the data and size of the data. Private key generator generates the private key for the client, it contains the client unique id. Client's data is transferred to combiner; the combiner distributes the data according to the size and type of data. Verifier sends the challenges to the combiner, the combiner transfer the challenge to the respected cloud. Afterwards, combiner aggregates the result and check whether it is valid or not. If it is valid, allow clients to store the data in multi cloud. In the phase Extract, PKG creates the private key for the client. The client creates the block-tag pair and uploads it to combiner. The combiner distributes the block-tag pairs to the different cloud servers according to the storage metadata. The verifier sends the challenge to combiner and the combiner distributes the challenge query to the corresponding cloud servers according to the storage metadata. The cloud servers respond the challenge and the combiner aggregates these responses from the cloud servers. The combiner sends the aggregated response to the verifier. Finally, the verifier checks whether the aggregated response is valid. The concrete ID-DPDP construction mainly comes from the signature, provable data possession and distributed computing. The signature relates the client's identity with his private key. Distributed computing is used to store the client’s data on multi-cloud servers. At the same time, distributed computing is also used to combine the multi-cloud servers’ responses to respond the verifier’s challenge. Based on the provable data possession protocol, the ID-DPDP protocol is constructed by making use of the signature and distributed computing.

II. MOTIVATION

To provide security first PDP scheme is proposed but due to I/O burden on the cloud. Secondly SPDP scheme is proposed it removes the limitation of I/O burden of first scheme but this scheme is unsuitable for third party verification. Then DPDP scheme is proposed but it takes very large time in integrity verification. After then CPOR Scheme is proposed the Limitation of this work is a Lack of some security issues for large files Lastly CPDP scheme is proposed which removes all the flaws of all the scheme for integrity verification One of the most important and most attention issues, that is in the cloud environment, servers within the data storage with security and integrity verification in this, we give an overview of our motivation in constructing CPDP. Our motivation is based on the following challenging questions that need to be addressed, which help us define our objectives in this paper.

III.RELATED WORK

RDPC allows a client that has stored data at a public cloud server (PCS) to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. In order to achieve secure RDPC implementations, Ateniese et al. proposed a provable data possession (PDP) paradigm [1] and designed two provably-secure PDP schemes based on the difficulty of large integer factoring. They refined the original paradigm and proposed a dynamic PDP scheme in [2] but their proposal does not support the insert operation. In order to solve this problem, Erway et al. proposed a full-dynamic PDP scheme by employing an authenticated flip table [3].Following Ateniese et al.’s pioneering work, researchers devoted great efforts to RDPC with extended models and new protocols [4], [5], [6], [7], [8], [9], [10], [11], [12]. One of the variations is the proof of retrievability (POR), in which a data storage server cannot only prove to a verifier that he is actually storing all of a client’s data, but also it can prove that the users can retrieve them at any time. This is stronger than the regular PDP notion. Shacham presented the first POR schemes with provable security. The state of the art can be found in [11] but few POR protocols are more efficient than their PDP counterparts. The challenge is to build POR systems that are both efficient and provably secure [14]. Note that one of benefits of cloud storage is to enable universal data access with independent geographical locations. This implies that the end devices may be mobile and limited in computation and storage. Regular RDPC protocols are more suitable for cloud users equipped with mobile end devices. Our IDRDPC architecture and protocol are based on the PDP model.

III. THE PROPOSED ID-DPDP PROTOCOL

In this section, we present an efficient ID-DPDP protocol. It is built from bilinear pairings which will be briefly reviewed below.

A. Bilinear pairings Let G1 and G2 be two cyclic multiplicative groups with the same prime order q. Let e : G1 × G1 → G2 be a bilinear map [25] which satisfies the following properties:

1) Bilinearity: ∀g1, g2, g3 ∈ G1 and a, b ∈Zq,

e(g1, g2g3) = e(g2g3, g1) = e(g2, g1)e(g3, g1) e(g1 a , g2 b) = e(g1, g2) ab

2) Non-degeneracy: ∃g4, g5 ∈ G1 such that e(g4, g5) 6= 1G2 .

3) Computability: ∀g6, g7 ∈ G1, there is an efficient algorithm to calculate e(g6, g7).

In the paper, the chosen group G1 satisfies that CDH problem is difficult but DDH problem is easy. The

DDH problem can be solved by making use of the bilinear pairings. Thus, (G1, G2) are also defined as GDH (Gap Diffie-Hellman) groups.

B. The Concrete ID-DPDP Protocol This protocol comprises four procedures: Setup, Extract, TagGen, and Proof. Its architecture can be depicted . The figure can be described as follows: 1. In the phase Extract, PKG creates the private key for the client. 2. The client creates the block-tag pair and uploads it to combiner. The combiner distributes the block-tag pairs to the different cloud servers according to the storage metadata. 3. The verifier sends the challenge to combiner and the combiner distributes the challenge query to the corresponding cloud servers according to the storage metadata. 4. The cloud servers respond the challenge and the combiner aggregates these responses from the cloud servers. The combiner sends the aggregated response to the verifier. Finally, the verifier checks whether the aggregated response is valid.

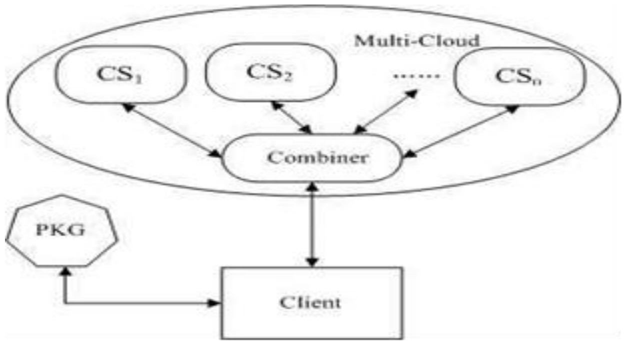
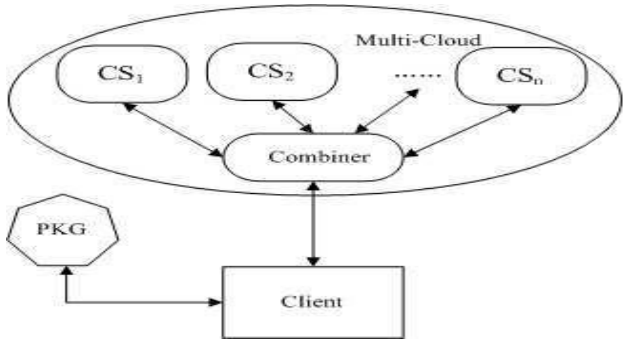


Fig. 1. The System Model of ID-DPDP



1) Client: an entity, which has massive data to be stored on the multi-cloud for maintenance and computation, can be either individual consumer or corporation.

2) CS (Cloud Server): an entity, which is managed by cloud service provider, has significant storage space and computation resource to maintain the clients' data.

3) 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.

4) PKG (Private Key Generator): an entity, when receiving the identity, it outputs the corresponding private key.

IV.CONTRIBUTIONS

In identity-based public key cryptography, this paper focuses on distributed provable data possession in multi-cloud storage. The protocol can be made efficient by eliminating the certificate management. We propose the new remote data integrity checking model: IDDPDP. The system model and security model are formally proposed. Then, based on the 342 bilinear pairings, the concrete ID-DPDP protocol is designed. In the random oracle model, our ID-DPDP 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 ID-DPDP protocol can realize private verification, delegated verification and public verification.

V.CONCLUSION

We presented the construction of an efficient PDP scheme for distributed cloud storage. Based on homomorphic verifiable response and hash Index hierarchy, we have proposed a cooperative PDP scheme to support dynamic scalability on multiple storage servers. We also showed that our scheme provided all security properties required by zero knowledge interactive proof system, so that it can resist various attacks even if it is deployed as a public audit service in clouds. Furthermore, we optimized the probabilistic query and periodic verification to improve the audit performance. Our experiments clearly Demonstrated that our approaches only introduce a small amount of computation and communication overheads. Therefore, our solution can be treated as a new candidate for data integrity verification in outsourcing data storage systems. As part of future work, we would extend our work to explore more effective CPDP constructions. Finally, it is still a challenging problem for the generation of tags with the length irrelevant to the size of data blocks. We would explore such a issue to provide the support of variable-length block verification

REFERENCES

1. 1.Yan Zhu, Hongxin Hu, Gail-JoonAhn, Mengyang Yu, ―Cooperative Provable Data Possession for Integrity Verification in Multicloud Storage‖, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 23, NO. 12, DECEMBER 2012.
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,‖ Proc. 14th ACM Conf. Computer and Comm. Security (CCS'07), pp. 598-609, 2007.
3. A. Juels and B.S.K. Jr., ―Pors: Proofs of Retrievability for Large Files,‖ Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 584-597, 2007.
4. 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.
5. G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, ―Scalable and Efficient Provable Data Possession,‖ Proc. Fourth Int'l Conf. Security and Privacy in Comm. Netowrks (SecureComm '08), pp. 1-10, 2008.
6. C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, ―Dynamic provable data possession‖, In CCS '09, pp. 213-222, April 24,2012.
7. Feifei Liu, DavuGu, HainingLu,‖An Improved Dynamic Provable Data Possession‖, Proceedings of IEEE CCIS2011, pp 290-295, 2011.
8. Zhifeng Xiao and Yang Xiao, ―Security and Privacy in Cloud Computing‖, The University of Alabama, Tuscaloosa, 24 March 2012.
9. Venkatesa Kumar V, Poornima G, ―Ensuring Data Integrity in Cloud Computing‖, Journal of Computer Applications ISSN: 0974 – 1925, Volume-5, Issue EICA2012-4, February 10, 2012.
10. 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,‖ Proc. ACM Symp. Applied Computing, pp. 1550- 1557, 2011.
11. Y. Zhu, H. Hu, G.-J.Ahn, Y. Han, and S. Chen, ―Collaborative Integrity Verification in Hybrid Clouds,‖ Proc. IEEE Conf. Seventh Int'l Conf. Collaborative Computing: Networking, Applications.
12. 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.

Guide Profile:

P.SRINIVASULU working as Associate Professor &HOD in Department of Computer Science and Engineering at MalinenikLakshmaiahEngineering College(MLEC), Singarayakonda. He completed his M.E in year 2008. His research area includes Data Stream Mining and Cloud Computing.

Student Profile:

Mr.V.RAKESH was born in AndhraPradesh,India. He recevivedB.tech Degree from JNTU Hyderabad ,Malinenilakshmaiah Engineering college prakasam district I am pursuing M.Tech Degree in CSE from JNTU Kakinada .