ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 12, March 2016

A REGENERATING POLICY BASED SECURED CLOUD STORAGE SCHEME USING PROXY

Manikandan.R [1], Mohanapriya.M [2]

1. P.G. Student, Dept. of MCA, VSB Engineering College, Karur, Tamilnadu, India
2. HoD, Dept. of MCA, VSB Engineering College, Karur, Tamilnadu, India

Abstract: **To protect outsourced data in cloud storage against corruptions, adding fault tolerance to cloud storage together with data integrity checking and failure reparation becomes critical. Recently, regenerating codes have gained popularity due to their lower repair bandwidth while providing fault tolerance. Existing remote checking methods for regenerating-coded data only provide private auditing, requiring data owners to always stay online and handle auditing, as well as repairing, which is sometimes impractical. In this paper, we propose a public auditing scheme for the regenerating-code-based cloud storage. To solve the regeneration problem of failed authenticators in the absence of data owners, we introduce a proxy, which is privileged to regenerate the authenticators, into the traditional public auditing system model. Moreover, we design a novel public verifiable authenticator, which is generated by a couple of keys and can be regenerated using partial keys. Thus, our scheme can completely release data owners from online burden. In addition, we randomize the encode coefficients with a pseudorandom function to preserve data privacy. Extensive security analysis shows that our scheme is provable secure under random oracle model and experimental evaluation indicates that our scheme is highly efficient and can be feasibly integrated into the regenerating-code-based cloud storage.**

Key Words: Cloud Storage, regenerating codes , public auditing, privacy preserving, proxy.

1. INTRODUCTION

A privacy-preserving public auditing system for data storage security in cloud computing in this the homomorphic linear authenticator and random masking to guarantee that the TPA[1] would not learn any knowledge about the data content stored on the cloud server during the efficient auditing process. It not only eliminates the burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage. Using cloud storage, usre can remotly store their data and enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources, without the burden of local data storage and maintenance. However, the fact the user on longer have physical possession of the outsourced data makes the data integrity protection in cloud computing a formidable task, especially for the users with constrained computing resource. Enabling public audit ability for cloud storage is of critical importance so that user can resort to a third party auditor (TPA) to check the integrity of outsourced data and be worry-free. To securely introduce an effective TPA, the auditing process should bring in no new vulnerabilities towards user data privacy, and introduce no additional online burden to user. Here the a secure cloud storage system supporting privacy-preserving public auditing is proposed. Data integrity maintenance is the major objective in cloud storage. It includes audition using TPA for unauthorized access. This work implements protecting the data and regeneration of data if someone mishandles it. This

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 12, March 2016

job will be assigned to a Proxy server. The data of the users will be stored in public and private area of the cloud. So that only public cloud data will be accessed by user and private cloud will remain more secured. Once any unauthorized modification is made, the original data in the private cloud will be retrieved by the Proxy server and will be returned to the user. Every data stored in the cloud will be generated with a Hash value using Merkle Hash Tree technique. So modification in content will make changes in the Hash value of the document as well. Proxy also perform signature delegation work by generating private and public key for every user using OEAP Algorithm so that the security will be maintained.

2 LITERATURE SURVEY

C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," presented privacy-preserving public auditing system for data storage security in Cloud Computing.

C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," proposed that a secure cloud storage system supporting privacy-preserving public auditing.

K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," proposed an efficient and inherently secure dynamic auditing protocol. It protects the data privacy against the auditor by combining the cryptography method with the bilinearity property of bilinear paring, rather than using the mask technique.

C. Wang, Q. Wang, K. Ren, N. Cao, and W. Lou, "Toward secure and dependable storage services in cloud computing," Proposed flexible distributed storage integrity auditing mechanism, utilizing the homomorphic token and distributed erasure-coded data. The proposed design allows users to audit the cloud storage with very lightweight communication and computation cost. The auditing result not only ensures strong cloud storage correctness guarantee, but also simultaneously achieves fast data error localization, i.e., the identification of misbehaving server.

The problem of remote data checking for integrity was first proposed in [26] and [27]. Then Ateniese et al. [2] and Juels and Kaliski [3] gave rise to the same notions provable data possession (PDP) and proof of retrievability (POR), respectively. Ateniese et al. [2] proposed a formal definition of the PDP model for ensuring possession of files on untrusted storage, introduced the concept of RSA-based homomorphic tags and adviced randomly sampling a some blocks of the file. In their subsequent work [28], they presented a dynamic version of the prior PDP scheme based on MAC, which permits very basic block operations with limited functionality but block insertions. At the same time, Erway et al. [29] gave a formal framework for dynamic PDP and provided the first fully dynamic solution to support provable updates to stored data using rank-based authenticated skit lists and RSA trees. To improve the efficiency of dynamic PDP, Wang et al. [30] Presented a new method which uses merkle hash tree to support fully dynamic data. To release the data owner from online burden for verification, [2] considered the public auditability in the PDP model for the first time. However, their variant protocol exposes the linear combination of samples and thus gives no data privacy guarantee. Then Wang et al. [14], [15] proposed a random blind technique to address that problem in their BLS signature based public auditing scheme. Similarly, Worku et al. [31]

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 12, March 2016

introduced another privacy-preserving method, which is more efficient since it avoids involving a computationally intensive pairing operation for the sake of data blinding. Yang and Jia [9] presented a public PDP scheme, where the data privacy is provided through combining the cryptography method with the bilinearity property of bilinear pairing. [16] used random mask to blind data blocks in error-correcting coded data for privacy-preserving auditing with TPA. Zhu et al. [10] presented a formal framework for interactive provable data possession (IPDP) and a zero-knowledge IPDP solution for private clouds. Their ZK-IPDP protocol supports fully data dynamics, public verifiability and is also privacy-preserving against the verifiers.

In this paper we are going to propose a public auditing scheme for the regenerating code based cloud storage. To obtain solution for regeneration problem of failed authenticators in the absence of data holders, we make a proxy, which is privileged to regenerate the authenticators, in the traditional public auditing system model. We also design a novel public verifiable authenticator, which is made by some keys. Thus, this scheme can almost release data holders from online burden. We also randomize the encode coefficients with a pseudorandom function to sure data privacy. Extensive security analysis shows this scheme is secure and provable under random oracle model. Experimental evaluation model indicates that this scheme is highly efficient and can be feasibly integrated i regenerating cloud based storage.

2.1 Existing System:

Many mechanisms dealing with the integrity of outsourced data without a local copy have been proposed under different system and security models up to now. The most significant work among these studies are the PDP (*provable data possession*) model and POR (*proof of retrievability*) model, which were originally proposed for the single-server scenario by Ateniese *et al.* and Juels and Kaliski, respectively. Considering that files are usually striped and redundantly stored across multi-servers or multi-clouds, explore integrity verification schemes suitable for such multi-servers or multi-clouds setting with different redundancy schemes, such as *replication*, *erasure codes*, and, more recently, *regenerating codes*.Chen *et al.* and Chen and Lee separately and independently extended the single-server CPOR scheme to the regeneratingcode-scenario; designed and implemented a data integrity protection (DIP) scheme for FMSR-based cloud storage and the scheme is adapted to the thin-cloud setting.

2.1.1 Disadvantages of Existing System:

- They are designed for private audit, only the data owner is allowed to verify the integrity and repair the faulty servers.
- Considering the large size of the outsourced data and the user's constrained resource capability, the tasks of auditing and reparation in the cloud can be formidable and expensive for the users
- The auditing schemes in existing imply the problem that users need to always stay online, which may impede its adoption in practice, especially for long-term archival storage.

3. SYSTEM MODEL

In this paper, we focus on the integrity verification problem in regenerating-code-based cloud storage, especially with the functional repair strategy. To fully ensure the data integrity and save the users' computation resources as well as online burden, we propose a public auditing scheme for the

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 12, March 2016

regenerating-code-based cloud storage, in which the integrity checking and regeneration (of failed data blocks and authenticators) are implemented by a third-party auditor and a semi-trusted proxy separately on behalf of the data owner. Instead of directly adapting the existing public auditing scheme to the multi-server setting, we design a novel authenticator, which is more appropriate for regenerating codes. Besides, we "*encrypt*" the coefficients to protect data privacy against the auditor, which is more lightweight than applying the proof blind technique and data blind method. We design a novel homomorphic authenticator based on BLS signature, which can be generated by a couple of secret keys and verified publicly. The system has 4 major components. They are,

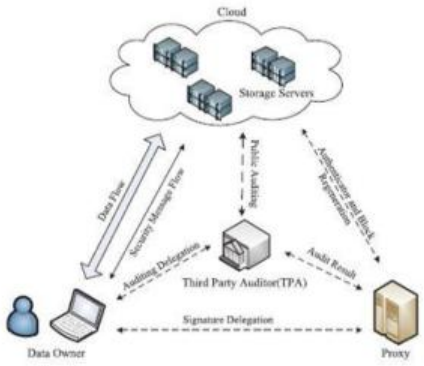

Fig.1 System Architecture

1.Data Owner: Who owns large amounts of data files to be stored in the cloud.

2.The Cloud: which are managed by the cloud service provider, provide storage service and have significant computational resources;

3.The Third Party Auditor (TPA): who has expertise and capabilities to conduct public audits on the coded data in the cloud, the TPA is trusted and its audit result is unbiased for both data owners and cloud servers;

Batch Auditing Module: With the establishment of privacy-preserving public auditing in Cloud Computing, TPA may concurrently handle multiple auditing delegations upon different users' requests. The individual auditing of these tasks for TPA can be tedious and very inefficient. Batch auditing not only allows TPA to perform the multiple auditing tasks simultaneously, but also greatly reduces the computation cost on the TPA side.

Data Dynamics Module: Hence, supporting data dynamics for privacy-preserving public risk auditing is also of paramount importance. Now we show how our main scheme can be adapted to build upon the existing work to support data dynamics, including block level operations of modification, deletion and insertion. We can adopt this technique in our design to achieve privacy-preserving public risk auditing with support of data dynamics.

4.Proxy Agent: who is semi-trusted and acts on behalf of the data owner to regenerate authenticators and data blocks on the failed servers during the repair procedure. A proxy agent acts on behalf of the data owner to regenerate authenticators and data blocks on the servers during the repair procedure. Notice that the data owner is restricted in computational and storage resources compared to other entities and may becomes off-line after the data upload procedure. Christo Ananth et al. [5] discussed about a system, In this proposal, a neural network approach is proposed for energy conservation routing in a wireless sensor network. Our designed neural network system has been successfully applied to our scheme of energy conservation. Neural network is applied to predict Most Significant Node and selecting the Group Head amongst the association of sensor nodes in

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 12, March 2016

the network. After having a precise prediction about Most Significant Node, we would like to expand our approach in future to different WSN power management techniques and observe the results. In this proposal, we used arbitrary data for our experiment purpose; it is also expected to generate a real time data for the experiment in future and also by using adhoc networks the energy level of the node can be maximized. The selection of Group Head is proposed using neural network with feed forward learning method. And the neural network found able to select a node amongst competing nodes as Group Head.

It generates signature using OAEP based key delegation which provides unique private and public key for each group registered in the cloud.

We have proposed auditing system model for Regenerating-Code-based cloud storage , which consist of four blocks: data owner which consist of large amount of data stored in the cloud; the cloud, which provides cloud services; provide storage service and have significant computational resources; the third party auditor (TPA) conducts public audits on the coded data in the cloud, its audit results are unbiased for both data owner and cloud servers; and proxy agent, who is semi trusted and acts on behalf of the data owner to regenerate authenticators and data blocks on the failed servers during the repair procedure. The data owner is restricted in computational and storage resources compared to other entities and may becomes off-line even after the data upload procedure. The proxy is supposed to be much more powerful than the data owner but less than the cloud servers in terms of computation and memory capacity, who would be always online. The periodic auditing and accidental repairing is used to save resources and online burden. The data owners resort to the TPA for integrity verification and delegate the reparation to the proxy. As compare to the traditional public auditing system model, our system model involves an additional proxy agent. In order to reveal the rationality of our design, we consider a scenario: A company employs a commercial regenerating-code-based public cloud and provides long-term archival storage service for its staffs, the staffs are equipped with low end computation devices (e.g., Laptop PC, Tablet PC, etc.) and will be frequently off-line. For public data auditing, the company relies on a trusted third party organization to check the data integrity; Similarly, to release the staffs from heavy online burden for data and authenticator regeneration, the company supply a powerful workstation (or cluster) as the proxy and provide proxy reparation service for the staffs' data.

A. Definitions of Auditing Scheme

Our auditing scheme consists three procedures: Setup, Audit, Repair.

Setup: Data owner used this procedure is to initialize our auditing scheme.

Audit: The cloud servers and TPA interact with one another to take a random sample on the blocks and check the data intactness in this procedure.

Repair: In the absence of the data owner, the proxy interacts with the cloud servers during this procedure to repair the wrong server detected by the auditing process.

B. Design Goals

To correctly and efficiently verify the integrity of data and keep the stored file available for cloud storage, our proposed auditing scheme should achieve the following properties:

Public Auditability: to allow TPA to verify the intactness of the data in the cloud on demand

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 12, March 2016

without introducing additional online burden to the data owner.

Storage Soundness: to ensure that the cloud server can never pass the auditing procedure except when it indeed manage the owner's data intact.

Privacy Preserving: to ensure that neither the auditor nor the proxy can derive users' data content from the auditing and reparation process.

Authenticator Regeneration: the authenticator of the repaired blocks can be correctly regenerated in the absence of the data owner.

Error Location: to ensure that the wrong server can be quickly indicated when data corruption is detected.

C. Security Analysis

Correctness: There are two verification process in this scheme, one for spot checking within the Audit phase and another for block integrity checking within the Repair phase.

Soundness : We say that our auditing protocol is sound if any cheating server that convinces the verification algorithm that it is storing the coded blocks and corresponding coefficients is actually storing them.

Regeneration-Unforgeable : Noting that the semi-trusted proxy handles regeneration of authenticators in our model, we say our authenticator is regeneration-unforgeable.

Resistant to Replay Attack: Our public auditing scheme is resistant to replay attack mentioned in [7], since the repaired server maintains identifier η_ which is different with the corrupted.

D Evaluation

Our proposed mechanism and makes a comparison with another remote data checking schemes [7], [8] for regenerating coding based cloud storage. We focus on evaluating the performance of our privacy preserving public audit scheme during the Setup, Audit and Repair procedure.

4 CONCLUSION AND FUTURRE SCOPE

In this paper, we present a public auditing scheme for the regenerating-code-based cloud storage system, where the data owners are privileged to delegate TPA for their data validity checking. To provide security to the original data privacy against the TPA, we randomize the coefficients in the starting rather than applying the blind technique within the auditing process. Data owner cannot always stay online always, in order to keep the storage available and verifiable after a malicious corruption, we present a semi-trusted proxy into the system model and give a privilege for the proxy to maintain the reparation of the coded blocks and authenticators. To better appropriate for the regenerating-code-scenario, we design our authenticator based on the BLS signature. This authenticator can be easily generated by the data owner at the same time with the encoding procedure. Extensive analysis provides that our scheme is provable secure, and the performance evaluation shows that our scheme is highly efficient and can be feasibly integrated into a regenerating-code-based cloud storage system.

REFERENCES

[1] M. Armbrust et al., "Above the clouds: A Berkeley view of cloud computing," Dept. Elect. Eng. Comput. Sci., Univ. California, Berkeley, CA, USA, Tech. Rep. UCB/EECS-2009-28, 2009.

[2] G. Ateniese et al., "Provable data possession at untrusted stores," in Proc. 14th ACM Conf. Comput. Commun. Secur. (CCS), New York, NY, USA, 2007, pp. 598–609.

[3] A. Juels and B. S. Kaliski, Jr., "PORs: Proofs of retrievability for large files," in Proc. 14th ACM

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 12, March 2016

Conf. Comput. Commun. Secur., 2007, pp. 584 597.

[4] R. Curtmola, O. Khan, R. Burns, and G. Ateniese, "MR-PDP: Multiple-replica provable data possession," in Proc. 28th Int. Conf. Distrib. Comput. Syst. (ICDCS), Jun. 2008, pp. 411–420.

[5] Christo Ananth, A.Nasrin Banu, M.Manju, S.Nilofer, S.Mageshwari, A.Peratchi Selvi, "Efficient Energy Management Routing in WSN", International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE), Volume 1, Issue 1, August 2015,pp:16-19

[6] J. He, Y. Zhang, G. Huang, Y. Shi, and J. Cao, "Distributed data possession checking for securing multiple replicas in geographically dispersed clouds," J. Comput. Syst. Sci., vol. 78, no. 5, pp. 1345–1358, 2012.

[7] B. Chen, R. Curtmola, G. Ateniese, and R. Burns, "Remote data checking for network coding-based distributed storage systems," in Proc. ACM Workshop Cloud Comput. Secur. Workshop, 2010, pp. 31–42.

[8] H. C. H. Chen and P. P. C. Lee, "Enabling data integrity protection in regenerating-coding-based cloud storage: Theory and implementation," IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 2, pp. 407–416, Feb. 2014.

[9] K. Yang and X. Jia, "An efficient and secure dynamic auditing protocol for data storage in cloud computing," IEEE Trans. Parallel Distrib. Syst., vol. 24, no. 9, pp. 1717–1726, Sep. 2013.

[10] Y. Zhu, H. Hu, G.-J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multicloud storage," IEEE Trans. Parallel Distrib. Syst., vol. 23, no. 12, pp. 2231–2244, Dec. 2012.

[11] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh, "A survey on network codes for distributed storage," Proc. IEEE, vol. 99, no. 3, pp. 476–489, Mar. 2011.

[12] H. Shacham and B. Waters, "Compact proofs of retrievability," in Advances in Cryptology. Berlin, Germany: Springer-Verlag, 2008, pp. 90–107.

[13] Y. Hu, H. C. H. Chen, P. P. C. Lee, and Y. Tang, "NCCloud: Applying network coding for the storage repair in a cloud-of-clouds," in Proc. USENIX FAST, 2012, p. 21.

[14] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing," in Proc. IEEE INFOCOM, Mar. 2010, pp. 1–9.