ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Secure Auditing for Regenerating Code Based Cloud Storage

Madhura.H.V, Dr. Thippeswamy

Department of Computer Science & Engineering, VTU PG Center, Mysore, India

Department of Computer Science & Engineering, VTU PG Center, Mysore, 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. This proposed system implementing Regenerating-code-based cloud storage. The system includes a proxy who is work as behalf of data owner, when data owner is offline or repair condition.

KEYWORDS: component; formatting; style; styling; insert (key words)

I. INTRODUCTION

Cloud computing is one of the emerging technologies that aims to provide on-demand use, device and location independence over the Internet, efficient share of resources, and maintenance. Cloud storage enables users to store and access the data in the remote cloud through the Internet regardless of location and time and it is an important service of cloud computing. Although this new paradigm of data hosting service promises more secure and reliable environment, it also brings new security vulnerability to the
users. Users' outsourced data can be lost or corrupted due to the inside and outside attacks, or system failures. However, cloud service providers might be dishonest. For example, they could discard the data that has been rarely accessed for monetary reasons, or they might even hide data loss incidents in order to maintain their reputations. Therefore, users need a way to check that their outsourced data are correctly stored in the cloud. To solve the problem of data integrity checking, many remote data auditing is used.

As cloud storage services have been widely adopted, the third party auditor may receive many requests from multi-users, even in multi-clouds. Thus, to improve the efficiency of the auditor, several batch auditing schemes have been proposed, which allow the auditor to simultaneously handle multiple
auditing delegations from different users . In large-scale cloud storage systems, batch auditing for multi-users in multi-clouds is essential to improve the auditing performance. In this paper, we give an survey of batch auditing schemes with public auditability. Firstly, we describe the concept of provable data possession-based remote data auditing as well as its system model and taxonomy. Then we review seven batch auditing schemes and compare them in terms of their features and performance. Furthermore, we introduce open research challenges with regard to batch auditing.

II. RELATED WORK

D Siva Chidambaram et.al [1] to solve the re-generating problem during failed authenticators in the absence of data owners, we introduce a proxy, which acts as a sub data owner to regenerate the authenticators, into the traditional public auditing system model. In addition, we design a public verifiable authenticator, which is generated by a couple of keys and can be regenerated using partial keys. Because of public auditing, our scheme can completely release online burden of data owners. To preserve data privacy coefficients are randomized and encoded with the pseudo-random functions. Extensive security analysis shows data are protected and our scheme is highly efficient, provable and can be practically integrated into the regenerating-code-based cloud storage.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Yongjun Ren [2] et.al Cloud computing is a promising computing model that enables convenient and on-demand network access to a shared pool of configurable computing resources. In cloud computing, data owners host their data on cloud servers and data consumers can access the data from cloud servers. This new paradigm of data storage service also introduces new security challenges, because data owners and data servers have different identities and different business interests. Therefore, an independent auditing service is required to make sure that the data is correctly hosted in the Cloud. However, the existing solutions are not specific to the multimedia data. Moreover copyright protection is not provided.

Sooyeon Shin [3] et.al present a survey of remote auditing schemes in the literature. We explain the concept of remote data auditing, and then describe system model and taxonomy of remote data auditing schemes, especially focusing on provable data possession schemes which support public audit ability and batch verification. We review seven batch auditing schemes and compare them in terms of their features and performance. Finally, we introduce some challenging issues in the design of efficient batch auditing.

Chunxiang Xu, [4] point out the security flaw existing in the scheme. An adversary, who is on-line and active, is capable of modifying the outsourced data arbitrarily and avoiding the detection by exploiting the security flaw. To fix this security flaw, we further propose a secure and efficient privacy-preserving public auditing scheme, which makes up the security flaw of Work et al.'s scheme while retaining all the features. Finally, we give a formal security proof and the performance analysis, they show the proposed scheme has much more advantages over the Worku et al.'s scheme.

New public[5] auditing scheme for regenerating-code-based cloud storage, to solve the regeneration criticality problems of failed authenticators in the absence of data owners; to introduce a proxy, which is legally protected to regenerate the authenticators, into the traditional public auditing system model. Moreover, to design a novel public verifiable authenticator is generated by a couple of keys and using partial key for regeneration. Thus, our scheme can completely free that the data owners from online burden. In addition, our system using randomized coefficient which is encoded with a pseudorandom function to preserve data protection. Our scheme is to feasibly integrate into the regenerating-code-based cloud storage and it should be highly efficient.

III. PROPOSED SYSTEM

We consider the auditing system model for Regenerating- Code-based cloud storage as Figure 1, which involves four entities: the data owner, who owns large amounts of data files to be stored in the cloud; the cloud, which are managed by the cloud service provider, provide storage service and have significant computational resources; 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; and a 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. Notice that 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, who would always be online, is supposed to be much more powerful than the data owner but less than the cloud servers in terms of computation and memory capacity. To save resources as well as the online burden potentially brought by the periodic auditing and accidental repairing, the data owners resort to the TPA for integrity verification and delegate the reparation to the proxy.

Here we use efficient access permission for each file based on the user attributes. When data owner uploading a file to cloud server, he can set access permissions such as read, download and modify based on user attributes (such as user's designation student, professor, etc).

Also here we use CP-ABE encryption scheme for encrypting the data.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

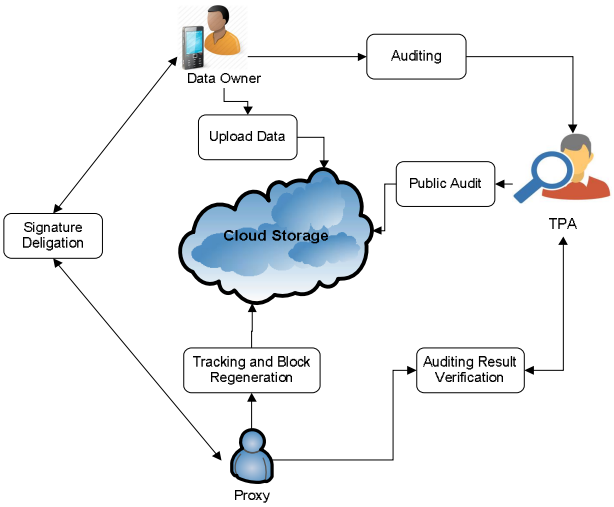

Figure 1:Architecture of Proposed System

A. File Upload and Security

In this module is used to upload the files present in user system and CP-ABE algorithm is used for file encryption. In such encryption scheme, an identity is viewed as a set of descriptive attributes, and decryption is possible if a descriptor's identity has some overlaps with the one specified in the ciphertext.

The file is present in the cloud is downloaded and use the CP-ABE algorithm for file decryption. In the CP-ABE, ciphertexts are created with an access structure, which specifies the encryption policy, and private keys are generated according to users attributes. A user can decrypt the ciphertext if and only if his attributes in the private key satisfy the access tree specified in the ciphertext. By doing so, the encrypter holds the ultimate authority about the encryption policy. Also, the already issued private keys will never be modified unless the whole system reboots.

We introduce how to accomplish the full anonymity in AnonyControl to outlines the completely unknown benefit control plan AnonyControl-F. The Key Generate calculation is the main part which spills individual data to every trait power. After accepting the characteristic key request with the trait esteem, the attribute authority will create $H(att(i))r_i$ and sends it to the requester where $att(i)$ is the attribute value and ri is an random number for that characteristic. The credit quality is unveiled to the power in this stride. We can add 1-out-of-n OT to prevent this leakage.

We let each authority be in charge of all attributes belonging to the same category. . For each attribute category, suppose there are k possible attribute values then one requester has at most one attribute value in one category. Upon the key request, the attribute authority can pick a random number ru for the requester and generates $H(att(i))r_u$ for all $i \in \{1, \ldots, k\}$.

After the attribute keys are prepared, the trait power and the key requester are occupied with a 1-out-of-k OT where the key requester needs to get one attribute key among k. By presenting the 1-out-of-k OT in our Key Generate calculation, the key requester accomplishes the right attribute key that he needs, however the attribute authority does not have any valuable data about authority is accomplished by the requester. At that point, the key requester accomplishes the full anonymity in our plan and regardless of what number of attribute authorities conspire; his identity data is kept secret.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

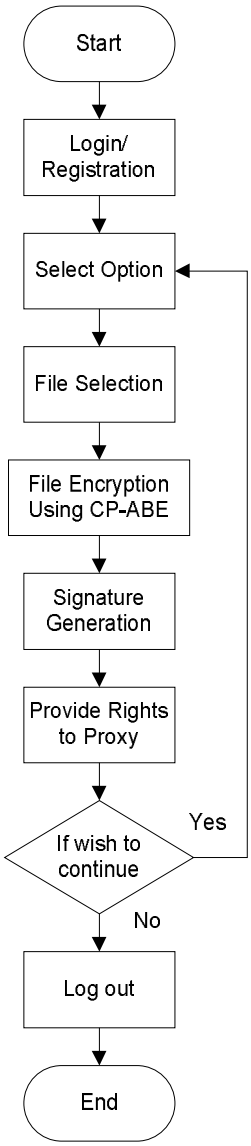

Figure 2: Uploading Algorithm

B. Cloud Storage

- Cloud storage stores the encrypted data of data owner. We can categorize the storage parts into several groups.
- Owner module is to upload their files using some access policy. First they get the public key for particular upload file after getting this public key owner request the secret key for particular upload file. Using that secret key owner upload their file.
- User module is used to help the client to search the file using the file id and file name .If the file id and name is incorrect means we do not get the file, otherwise server ask the public key and get the encrypted file. If user wants the decrypted file means user must have the secret key.

C. Regenerating Code

To correctly and efficiently verify the integrity of data an 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 without introducing additional online burden to the data owner.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

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.

IV. RESULT AND DISCUSSION

Experimental result analysis is the process of analyzing the output of experiments carried on the system. The verified application has been experimented with various inputs and the results are analyzed for its performance and accuracy. The detail analysis of the experiments and results obtained is explained below.

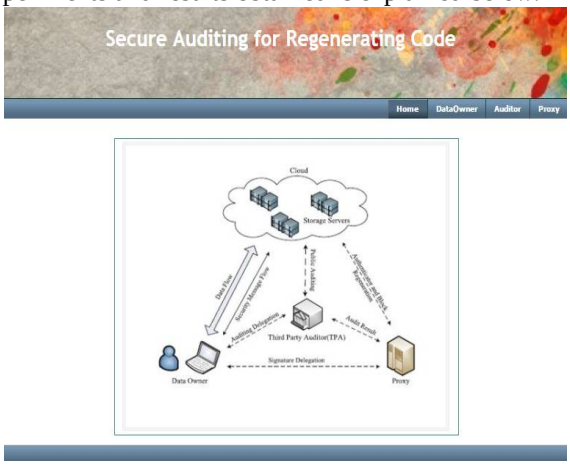

Figure 2:Home page

Figure 3:File upload page for data owner. Data is encrypted using

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Figure 4:Signature generation by proxy. Data owner must give a proxy signature to provide authority to proxy to act as data owner's behalf.

Figure 5:Successful Generation of Proxy Signature. Data owner provide the proxy signature to proxy and using the signature proxy can login on behalf of

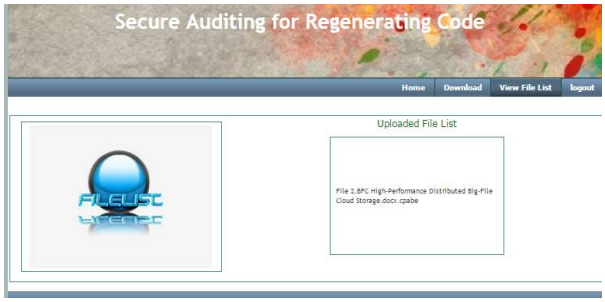

Figure 6: Proxy Home Page and View file list. From these file proxy select the file to be regenerated.

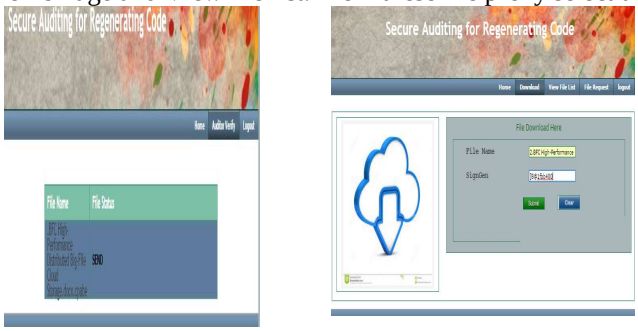

Figure 7: for the regeneration of data owner data. TPA must authenticate proxy by sending file signature. When signature verification is successful proxy can reconstruct the file data by regenerating code.

ISSN(Online) : 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

V. CONCLUSION

Proposed a brought together approach for consistence administration in cloud situations that backings the cloud administration customer's point of view utilizing agreeable administration to help the cloud administration purchasers in selecting legitimate cloud administrations. Dissimilar to past works that utilization unified design, we display a believability model supporting disseminated trust criticism appraisal and capacity. This validity display likewise recognizes dependable and malevolent trust inputs.

REFERENCES

[1] D Siva Chidambaram.V Pandarinathan.U ,Hariganesh.P Mathivanan.M Parthiban, IJRCS – Volume 3 Issue,Volume 03 Issue 01 Year 2016.
[2] Yongjun Ren, Jian Shen, Jin Wang,JiangXu, and Liming Fang, "Security Data Auditing based on Multifunction Digital Watermark for Multimedia File in Cloud Storage". Vol.9, No.9 (2014).
[3] Sooyeon Shin and Taekyoung Kwon, "A Survey of Public Provable Data Possession Schemes with Batch Verification in Cloud Storage".
[4] Chunxiang Xu , Yuan Zhang , Yong Yu , Xiaojun Zhang , Junwei Wen." An Efficient Provable Secure Public Auditing Scheme for Cloud Storage", KSII Transactions on Internet and Information Systems (TIIS). 2014. Nov, 8(11): 4226-4241.
[5] B.Lekha, Prof. P. Kirubanantham, "Secure Public Auditing With Network Coding Based Storage in Cloud Architecture", 2016 IJEDR | Volume 4, Issue 1 | ISSN: 2321-9939.
[6] .Satish Shelar and S.Y.Raut "Review On Regenerating Code Based Secure Cloud Storage Using Public Auditing", International Research Journal of Engineering and Technology, Volume: 02 Issue: 09, 2015.
[7] C.Jeevitha,T. Senthil Prakash and .C.Janani, "Privacy-Preserving Public Auditing for Regenerating-Code-Based Cloud Storage", International Journal of Research in Science and Technology, Vol. No. 6, 2016.
[8] Monjur Ahmed and Mohammad Ashraf Hossian, Cloud Computing and Security Issues in the Cloud, International Journal of network security and Its Application.
[9] V anto Vins, S. Umamageswari, P. Saranya, A survey on Regenerating code, International Journal of Scientific and Research Publication, November 2014.
[10] 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.
[11] C. Wang, S. S. M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for secure cloud storage," IEEE Trans. Comput., vol. 62, no. 2, pp. 362–375, Feb. 2013.