ISSN(Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

Implementation of Secure Cloud storage with Multi layer Design Appraoch

Uday Nalavade[1], Prof.Vina Lomte[2]

P.G. Student, Department of Computer Engineering RMD Sinhagad Institute of Technology, Warje Campus, India

Associate Professor, Department of Computer Engineering RMD Sinhagad Institute of Technology,

Warje Campus, India

ABSTRACT: Cloud storage auditing is viewed as an important service to verify the integrity of the data in public cloud. Current auditing protocols are all based on the assumption that the client's secret key for auditing is absolutely secure. However, such assumption may not always be held, due to the possibly weak sense of security and/or low security settings at the client. If such a secret key for auditing is exposed, most of the current auditing protocols would inevitably become unable to work. In this paper, we focus on this new aspect of cloud storage auditing. We investigate how to reduce the damage of the client's key exposure in cloud storage auditing, and give the first practical solution for this new problem setting. We formalize the definition and the security model of auditing protocol with key-exposure resilience and propose such a protocol. In our design, we employ the binary tree structure and the preorder traversal technique to update the secret keys for the client. We also develop a novel authenticator construction to support the forward security and the property of blockless verifiability. The security proof and the performance analysis show that our proposed protocol is secure and efficient.

KEYWORDS*:* Data Integrity, cloud secure Storage, key Management,

I. INTRODUCTION

Cloud computing can help enterprises improve the creation and delivery of IT solutions by providing them with access to services in a cost-effective and flexible manner [2]. Clouds can be classified into three categories, depending on their accessibility restrictions and the deployment model.

Public Cloud

Private Cloud

Hybrid Cloud

A public Cloud is made available in a pay-as-you-go manner to the general public users irrespective of their origin or affiliation. A private Cloud's usage is restricted to members, employees, and trusted partners of the organization. A hybrid Cloud enables the use of private and public Cloud in a seamless manner. Cloud computing applications span many domains, including business, technology, government,

ISSN(Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

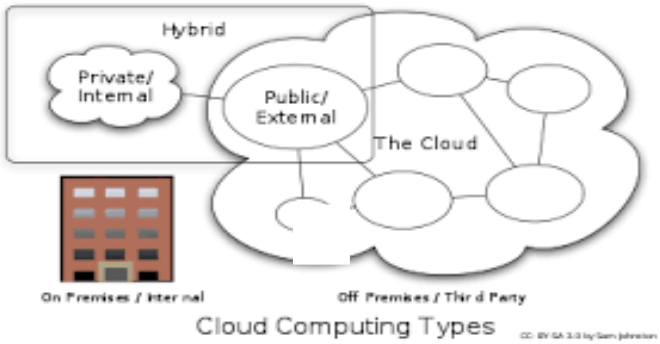

Figure 1: cloud Computing Types

care, smart grids, intelligent transportation networks, life sciences, disaster management, automation, data analytics, and consumer and social networks. Various models for the creation, deployment, and delivery of these applications as Cloud services have emerged. Cloud storage auditing is used to verify the integrity of the data stored in public cloud, which is one of the important security techniques in cloud storage. In recent years, auditing protocols for cloud storage have attracted much attention and have been researched intensively [16]. These protocols focus on numerous different characteristics of auditing, achieving high bandwidth and computation efficiency is one of the essential concerns. For that perseverance, the Homomorphic Linear Authenticator (HLA) technique that maintains block less verification is explored to diminish the overheads of computation and communication in auditing protocols, which allows the auditor to verify the integrity of the data in cloud without retrieving the whole data. Many cloud storage auditing protocols have been proposed based on this technique. In order to reduce the computational burden of the client, a third -party auditor (TPA) is introduced to help the client to periodically check the integrity of the data in cloud. However, it is possible for the TPA to get the client's data after it executes the auditing protocol multiple times. Auditing protocols in [9] and [10] are designed to ensure the privacy of the client's data in cloud. While all existing protocols focus on the faults or dishonesty of the cloud, they have o overlooked he possible weak sense of security and/or low security settings at the client. The procedure to deal with the client's secret key exposure for cloud storage auditing is a very important problem. It is focused here on how to reduce the damage of the clients key exposure in cloud storage auditing

II. RELATED WORK

Research [1] focuses on these protocols focus on several different aspects of auditing, and how to achieve high bandwidth and computation efficiency is one of the essential concerns. For that purpose, the Homomorphic Linear Authenticator (HLA) technique that supports block less verification is explored to reduce the overheads of computation and communication in auditing protocols, which allows the auditor to verify the integrity of the data in cloud without retrieving the whole data.

Research **[2]**The privacy protection of data is also an important aspect of cloud storage auditing. In order to reduce the computational burden of the client, a third-party auditor (TPA) is introduced to help the client to periodically check theintegrity of the data in cloud. However, it is possible for the TPA to get the client's data after it executes the auditing protocol multiple times

[3] have proposed an auditing protocol supporting fully dynamic data operations including modification, insertion and deletion.

Though many research works about cloud storage auditing have been done in recent years, a critical security problem—the key exposure problem for cloud storage auditing, has remained unexplored in previous researches. While

ISSN(Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

all existing protocols focus on the faults or dishonesty of the cloud, they have overlooked the possible weak sense of security and/or low security settings at the client.

Here author have presented [4] design and implementation of practical data integrity protection (DIP) scheme for a specific regenerating code. This is with preserving its intrinsic properties of fault tolerance and repair-traffic saving. DIP scheme is designed under a mobile Byzantine adversarial model. This provides enables a feasibility to client for verifying the integrity of random subsets of outsourced data against general or malicious corruptions. Mathematical models are used for further analysis of the security strengths of DIP scheme. We are able to prove that remote integrity checking can be feasibly integrated into regenerating codes in practical deployment. Limitation of the proposed work is that they are designed for a single-server setting. We pose the study of different possible corruption approaches as future work.

[4]Unfortunately, previous auditing protocols did not consider this critical issue of how to deal with the client's secret key exposure for cloud storage auditing, and any exposure of the client's secret auditing key would make most of the existing auditing protocols unable to work correctly

Though many research works about cloud storage auditing have been done in recent years, a critical security problem—the key exposure problem for cloud storage auditing, has remained unexplored in previous researches. While all existing protocols focus on the faults or dishonesty of the cloud, they have overlooked the possible weak sense of security and/or low security settings at the client.

[5]Unfortunately, previous auditing protocols did not consider this critical issue of how to deal with the client's secret key exposure for cloud storage auditing, and any exposure of the client's secret auditing key would make most of the existing auditing protocols unable to work correctly.

[6]on how to reduce the damage of the clients key exposure in cloud storage auditing. Our goal is to design a cloud storage auditing protocol with built-in key-exposure resilience. How to do it efficiently under this new problem setting brings in many new challenges to be addressed below. First of all, applying the traditional solution of key revocation to cloud storage auditing is not practical. This is because, whenever the client's secret key for auditing is exposed, the client needs to produce a new pair of public key and secret key and regenerate the authenticators for the client's data previously stored in cloud.

[7] goal is to design a practical auditing protocol with key-exposure resilience, in which the operational complexities of key size, computation overhead and communication overhead should be at most sub-linear to T. In order to achieve our goal, we use a binary tree structure to appoint time periods and associate periods with tree nodes by the pre-order traversal technique. The secret key in each time period is organized as a stack. In each time period, the secret key is updated by a forward-secure technique.

[8]The auditing protocol achieves key-exposure resilience while satisfying our efficiency requirements. As we will show later, in our protocol, the client can audit the integrity of the cloud data still in aggregated manner, i.e., without retrieving the entire data from the cloud.

Clouds' consumers [8] have to know in advance where their data will be resided and how will be segregated in order to avoid data leakage problems. In addition, lack of visibility about the way data is stored and secured, lead to a number of concerns have to be considered when moving to the cloud. Data centers are not stand alone; it has to be connected to other data centers. So, security and latency should be managed in a proper way.Compliance, security-breach audit and forensics are used to insure no one violates or attacks the security within the system.Encryption is often used to secure data is not trusted storage environment such

[9,10] security of protocol in the formalized security model, and justify its performance via concrete asymptotic analysis. Indeed, the proposed protocol only adds reasonable overhead to achieve the key-exposure resilience. We also show that our proposed design can be extended to support the TPA, lazy update and multiple sectors.

ISSN(Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

III. PROPOSED METHODOLOGY

Proposed system has been implemented in four layer approach as shown below

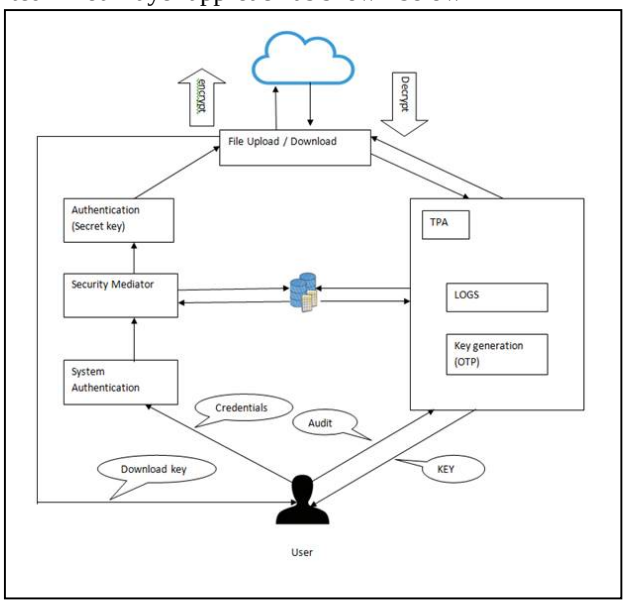

Figure 2: Proposed Methodology

proposed the framework for general development that help useranonymity and outsider open auditing.Here both substance have gotten impressive consideration recently.i.e., secure cloudstorage and secure system coding.Generally client first outsourcesits information to the cloud. In this then client intermittently performsan review on the honesty of outsourced information. For security reason usercan then check whether the verification came back from the cloud isvalid or not. This way to keep the information remains intact.This acquire a proof that the information has been altered which willpossibly cause some further activity, for example, legitimate activity or information recuperation yet that is consider here.

Entities Interacting in Research protocol

TPA :

TPA performs the main role of data integrity check. It performs activities like generating hash value for encrypted blocks received from cloud server, concatenating them and generates signature on it. It later compares both the signatures to verify whether the data stored on cloud is tampered or not. It verifies the integrity of data on demand of the users. The cloud server is used only to save the encrypted blocks of data.

Server : Uploads Data File to cloud and Performs Data Chunking and splitting of files .

Encryption module: AES DES and Key Management Procedure is been iomplmenetd.

Key Management procedure:

A dynamic Key management Algorithm has been implemented which Generates Secret key for Encrypting data from user.

Proposed Methodology has been as shown below

Layer 1

Key Generation: Run by client

Input: None.

Output: public key, secrete key, generator g.

ISSN(Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

Layer 2
Signature Generation: Run by client
Input: File Blocks F, secret key, generator g
Output: Set of signature

Layer 3
Generate Proof: Run by cloud storage server
Input: Subset of file blocks mi, coefficient i
Output: Proof P

Layer 4
VerifyProof: Run by TPA
Input: Proof P
Output: Boolean value (TRUE, FALSE)

Layer 5
ExecUpdate: Run by the server
Input: file F, set of signature, update query
Output: new file F, new set of signature, update proof.

Layer 6
VerifyUpdate: Run by the client.
Input: public key, update query, update proof
Output: Boolean value TRUE, FALSE, and signature

IV. EXPERIMENTAL RESULTS

Research activity depicted an orderly writing audit that directed to examine the present state of information about secure cloud storage, and a research goal to achieve data security and privacy. It distinguished essential studies as per review convention, and examined issues. To proof that the research performed here with is effective to reveal privacy issues. The implementation includes coverage of determined security issues from user side and TPA verification. It also solves determined ambiguity, hence causing the effective methodology that improves privacy preserving of data in cloud from two way .Higher level of data integrity with key management Data Block creation safe guard's data from attackTPA assurances data integrity.

V. CONCLUSION

The proposed system secures cloud from internal as well as network attack .In future Work AES and DES could be replaced with Higher Encryption Standards like SHA512 SHA312 etc.

REFERENCES

[1] Wang, Cong, et al. "Privacy-preserving public auditing for secure cloud storage." *IEEE transactions on computers* 62.2 (2013): 362-375.

[2] Gkantsidis, Christos, and Pablo Rodriguez. "Cooperative Security for Network Coding File Distribution." *INFOCOM*. Vol. 3. No. 2006. 2006.

[3] Yang, Kan, and Xiaohua Jia. "An efficient and secure dynamic auditing protocol for data storage in cloud computing." *IEEE transactions on parallel and distributed systems* 24.9 (2013): 1717-1726.

[4] Ahlswede, Rudolf, et al. "Network information flow." *IEEE Transactions on information theory* 46.4 (2000): 1204-1216.

[5] Zhao, Fang, et al. "Signatures for content distribution with network coding." *Information Theory, 2007. ISIT 2007. IEEE International Symposium on*. IEEE, 2007.

[6] Shah, Mehul A., Ram Swaminathan, and Mary Baker. "Privacy-Preserving Audit and Extraction of Digital Contents." *IACR Cryptology EPrint Archive* 2008 (2008): 186.

[7] Bessani, Alysson, et al. "DepSky: dependable and secure storage in a cloud-of-clouds." *ACM Transactions on Storage (TOS)* 9.4 (2013): 12.

ISSN(Online): 2319-8753
ISSN (Print) : 2347-6710

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 6, Issue 10, October 2017

[8] Chen, Henry CH, and Patrick PC Lee. "Enabling data integrity protection in regenerating-coding-based cloud storage: Theory and implementation." *IEEE transactions on parallel and distributed systems* 25.2 (2014): 407-416.

[9] Wang, Boyang, Baochun Li, and Hui Li. "Panda: Public auditing for shared data with efficient user revocation in the cloud." *IEEE Transactions on services computing* 8.1 (2015): 92-106.

[10] Joshi, Mahima, and Yudhveer Singh Moudgil. "Secure cloud storage." *International Journal of Computer Science & Communication Networks* 1.2 (2011): 171-175.

[11] Alysson Bessani et.al, "DEPSKY: Dependable and Secure Storage in a Cloud-of-Clouds", EuroSys'11, April 10–13, 2011, Salzburg, Austria. Copyright 2011 ACM 978-1-4503-0634-8/11/04.

[12] Henry C.H. Chen et.al, "NCCloud: A Network-Coding-Based Storage System in a Cloud-of-Clouds", IEEE TRANSACTIONS ON COMPUTERS, VOL. 63, NO. 1, JANUARY 2014

[13] Henry C.H. Chen et.al, "Enabling Data Integrity Protection in Regenerating-Coding-Based Cloud Storage: Theory and Implementation", IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 25, NO. 2, FEBRUARY 2014

[14] oyang Wang et.al, "Storing Shared Data on the Cloud via Security-Mediator", 2013 IEEE 33rd International Conference on Distributed Computing Systems, 1063-6927/13 $26.00 © 2013 IEEE DOI 10.1109/ICDCS.2013.60