Cloud Server Storage Security Using TPA

[I]K.Meenakshi, [II]Victo Sudha George
[I]Student (M.Tech), DR. MGR Educational and Research Institute, India
[II]Assisstant Professo, DR. MGR Educational and Research Institute

Abstract

In cloud computing, data is moved to a remotely located cloud server. Cloud server faithfully stores the data and return back to the owner whenever needed. Many users place their data in the cloud and so data integrity is very important issue in cloud storage. After moving the data to the cloud, owner hopes that their data and applications are in secured manner. But that hope may fail sometimes that is the owner's data may be altered or deleted. In this scenario, the user must download the data in order to validate it. If the outsourced data is very large files, such downloading to determine data integrity may become prohibitive in terms of increased cost of bandwidth and time, especially if frequent data checks are necessary. This project propose a method that, owner need not download the data or files to check the integrity and it provides the proofs that data is stored at a remote storage in cloud is not modified by anyone and there by integrity of the data is assured. Because data integrity ensured that data is of high quality, correct, consistent, and accessible. Data owner can resort to a Third Party Auditor (TPA) to check the integrity of data stored in cloud server.

Keywords

TPA, MHT, POR

I. Introduction

Cloud Computing is the long dreamed vision of computing as a utility, where users can remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. By data outsourcing, users can be relieved from the burden of local data storage and maintenance. However, the fact that users no longer have physical possession of the possibly large size of outsourced data makes the data integrity protection [1] in Cloud Computing a very challenging and potentially formidable task, especially for users with constrained computing resources and capabilities. Thus, enabling public auditability [6] for cloud data storage security is of critical importance so that users can resort to an external audit party to check the integrity of outsourced data when needed.

Third Party Auditor (TPA) is the secured one, which have the two fundamental requirements have to be met: 1) TPA should be able to efficiently audit the cloud data storage without demanding the local copy of data, and introduce no additional on-line burden to the cloud user; 2) The third party auditing process should bring in no new vulnerabilities towards user data privacy. In this project, we utilize and uniquely combine the public key based homomorphic authenticator with random masking to achieve the privacy-preserving public cloud data auditing system, which meets all above requirements. To support efficient handling of multiple auditing tasks, we further explore the technique of bilinear aggregate signature to extend our main result into a multi-user setting, where TPA can perform multiple auditing [8] simultaneously. Extensive security and performance analysis shows the proposed schemes are provably secure and highly efficient.

II. Objective

The objective of this project is to provide the security for the data that are stored in the cloud environment using privacy- preserving public auditing mechanism, and so that we can increase the security level [2] for the data that are stored in the cloud servers.

III. Literature Survey

Title

Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing

Description

Cloud Computing has been envisioned as the next-generation architecture of IT Enterprise. It moves the application software and databases to the centralized large data centers, where the management of the data and services may not be fully trustworthy. This unique paradigm brings about many new security challenges, which have not been well understood. This work studies the problem of ensuring the integrity of data storage in Cloud Computing. In particular, we consider the task of allowing a third party auditor (TPA), on behalf of the cloud client, to verify the integrity of the dynamic data stored in the cloud. The introduction of TPA eliminates the involvement of the client through the auditing of whether his data stored in the cloud is indeed intact, which can be important in achieving economies of scale for Cloud Computing. The support for data dynamics via the most general forms of data operation, such as block modification, insertion and deletion, is also a significant step toward practicality, since services in Cloud Computing are not limited to archive or backup data only. While prior works on ensuring remote data integrity often lacks the support of either public auditability or dynamic data operations, this paper achieves both. We first identify the difficulties and potential security problems of direct extensions with fully dynamic data updates from prior works and then show how to construct an elegant verification scheme for the seamless integration of these two salient features in our protocol design. In particular, to achieve efficient data dynamics, we improve the existing proof of storage models by manipulating the classic (MHT) Merkle Hash Tree construction for block tag authentication. To support efficient handling of multiple auditing tasks, we further explore the technique of bilinear aggregate signature to extend our main result into a multi-user setting, where TPA can perform multiple auditing tasks simultaneously. Extensive security and performance analysis show that the proposed schemes are highly efficient and provably secure.

Title

Privacy-Preserving Public Auditing for Secure Cloud Storage

Description

Using Cloud Storage, users can remotely 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 that users no longer have physical possession of the outsourced data makes the data integrity protection in Cloud Computing a formidable task, especially for users with constrained computing resources. Moreover, users should be able to just use the cloud storage as if it is local, without worrying about the need to verify its integrity. Thus, enabling public auditability for cloud storage is of critical importance so that users 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. In this paper, we propose a secure cloud storage system supporting privacy-preserving public auditing. We further extend our result to enable the TPA to perform audits for multiple users simultaneously and efficiently. Extensive security and performance analysis show the proposed schemes are provably secure and highly efficient. Our preliminary experiment conducted on Amazon EC2 instance further demonstrates the fast performance [2] of the design.

Title

PORs: Proofs of Retrievability for Large Files

Description

In this paper, we define and explore proofs of retrievability (PORs) [3]. A POR scheme enables an archive or back-up service (prover) to produce a concise proof that a user (verifier) can retrieve a target file F, that is, that the archive retains and reliably transmits file data sufficient for the user to recover F in its entirety. A POR may be viewed as a kind of cryptographic proof of knowledge (POK), but one specially designed to handle a large file (or bitstring) F. We explore POR protocols here in which the communication costs, number of memory accesses for the prover, and storage requirements of the user (verifier) are small parameters essentially independent of the length of F. In addition to proposing new, practical POR constructions, we explore implementation considerations and optimizations that bear on previously explored, related schemes. In a POR, unlike a POK, neither the prover nor the verifier need actually have knowledge of F. PORs give rise to a new and unusual security definition whose formulation is another contribution of our work. We view PORs as an important tool for semi-trusted online archives. Existing cryptographic techniques help users ensure the privacy and integrity of files they retrieve. It is also natural, however, for users to want to verify that archives do not delete or modify files prior to retrieval. The goal of a POR is to accomplish these checks without users having to download the files themselves. A POR can also provide quality-of-service guarantees, i.e., show that a file is retrievable within a certain time bound.

Title

Privacy-Preserving Audit and Extraction of Digital Contents

Description

A growing number of online services, such as Google, Yahoo!, and Amazon, are starting to charge users for their storage. Customers often use these services to store valuable data such as email, family photos and videos, and disk backups [4]. Today, a customer must entirely trust such external services to maintain the integrity of hosted data and return it intact. Unfortunately, no service is infallible. To make storage services accountable for data loss, we present protocols that allow a third party auditor to periodically verify the data stored by a service and assist in returning the data intact to the customer. Most importantly, our protocols are privacy-preserving, in that they never reveal the data contents to the auditor. Our solution removes the burden of verification from the customer, alleviates both the customer's and storage service's fear of data leakage, and provides a method for independent arbitration of data retention contracts.

IV. Existing System

The traditional cryptographic technologies for data integrity and availability, cannot work on the outsourced data. It is not a practical solution for data validation by downloading them due to the expensive communications, especially for large size files. Moreover, the ability to audit the correctness of the data in a cloud environment can be formidable and expensive for the cloud users. In the existing system, the correctness of the data in the cloud is being put at risk due to the following reasons. Although the infrastructures under the cloud [2] are much more powerful and reliable than personal computing devices, they are still facing the broad range of both internal (loss or destruction of data) and external (exposure of data to unauthorized users) threats for data integrity.

V. Proposed System

In the proposed system, we propose an effective and flexible distributed scheme with explicit dynamic data support to ensure the correctness of users' data in the cloud. In the proposed System, we are implementing the secure system namely Privacy preserving auditing. In this system, first the Data Owner will register with the Cloud Service Providers. During the registration phase the Public and Private will be generated for the Data Owner [1]. The Data Owner have to provide their Private key while updating their data in the Cloud Server. Using Merkel Hash Tree Algorithm the Cloud Server Split the data into batches. The Cloud Server will allow the Third Party Auditor (TPA) to audit the data that was Stored in the Cloud Server as requested by the User. The TPA will also audit multiple files.

VI. Comparison of Existing System With Proposed System

In existing system, there is no proper mechanism was implemented to audit the data that are stored in cloud servers and so security is also very low. Due to the poor data auditing mechanism in existing system, customer of the company also be reduced in commercial area of business.

But in proposed system, TPA audit the files as requested by data owner through a new mechanism called Merkel Hash Tree algorithm and so very high security also provided. And also,in proposed system, TPA can perform multiple auditing tasks simultaneously.

VII. System Architecture

System network architecture for cloud data storage is illustrated in Fig.7.1. Three different network entities[2] can be identified as follows.

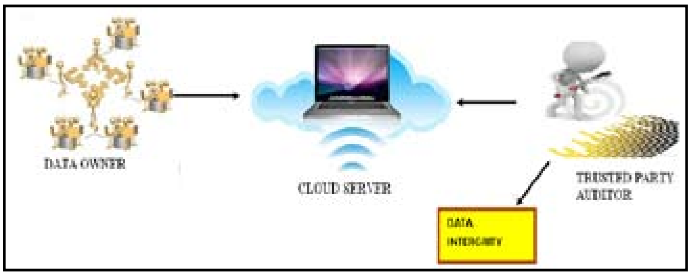


Fig. 1: Architecture of Cloud Server Storage

A. Data Owner (Client)

An entity, which has large data files to be stored in the cloud and relies on the cloud for data maintenance and computation, can be either individual consumers or organizations.

B. Cloud Storage Server (CSS)

An entity, which is managed by Cloud Service Provider (CSP), has significant storage space and computation resource to maintain the client's data.

C. Trusted (Third) Party Auditor (TPA)

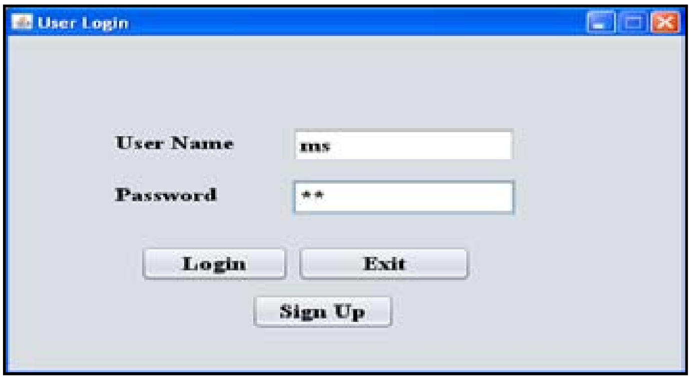
An entity, which has expertise and capabilities that clients do not have, is trusted to assess and expose risk of cloud storage services on behalf of the clients upon request. The data owner registered with cloud service provider and stores their data in cloud server by using the private key. To check the integrity of data [9] stored in cloud server, data owner sends request to Third Party Auditor(TPA).The TPA audit the files stored in cloud server by using top hash value [15] which it get from the data owner.

VIII. Modules

- Cloud Client-Server Registration
- Data Dynamics
- Merkle Hash Tree Algorithm
- Trusted Party Auditor
- Random Auditing

A. Cloud Client-Server Registration

Cloud Client-Server Registration is a module which used to create network which consist of Users and Server to communicate with each other. So that we are developing a Network which consist of Data Owner/ User and the Cloud Service Provider. To implement this module we'll design the User Interface frame for both of them. So that the Cloud Service Provider will monitor the Data Owner/ User's accessing information. Also the Cloud Service Provider [7] will have maintain the Database to Store and Retrieve the information of the Registered User's in the Cloud Network.


User Login

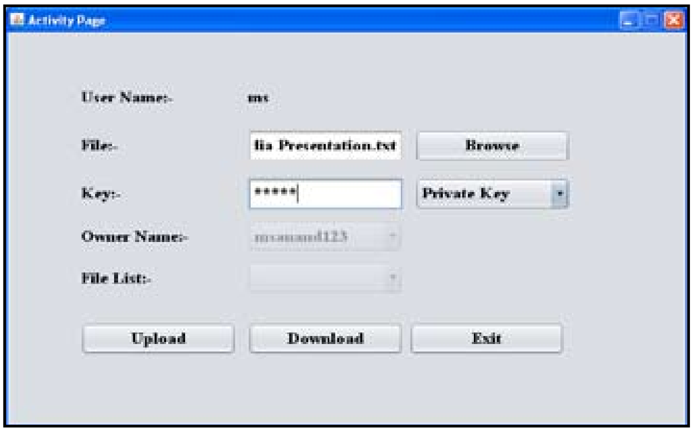
User Name: ms

Password: **

Fig. 2: Client Login To Server

B. Data Dynamics

In this module, the most general forms of data operation, such as data block insertion, deletion also a significant step can be performed by data owner using private key [2].


Activity Page

User Name:- ms

File:- lia Presentation.txt

Key:- *****

Private Key

Owner Name:- msanand123

File List:-

Fig. 2(a): Client Entering Private Key

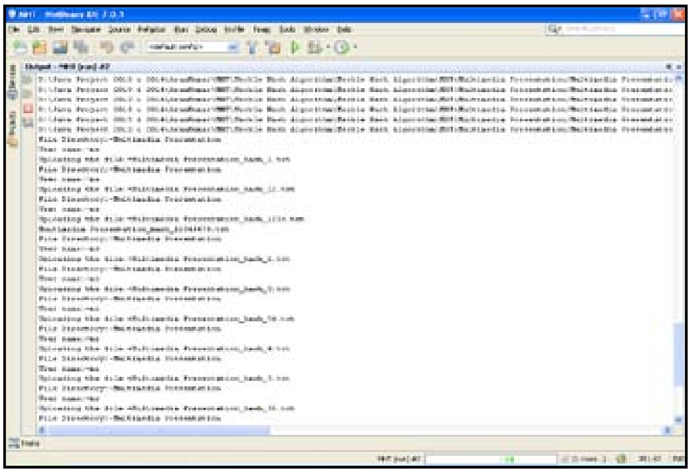


Fig. 2(b) File Uploading Into Cloud Server

C. Merkle Hash Tree Algorithm

Once the data owner provide the data to be audited, the Trusted Party Auditor will audit the data using Merkle Hash Tree algorithm. Using this algorithm the data will audited from the child node to root node [1]. First the data will splitted into two parts, then each of them will be splitted into the another two parts. Then each of the will be splited into another two parts (maximum upto eight parts like p1, p2, p3, p4, p5, p6, p7, p8).

Once splitted, it generates the hash value for all child nodes separately(h1, h2, h3, h4, h5, h6, h7, h8), and then the TPA will audit the data from Child node to root node. Started from child nodes, sequentially first two nodes (h1 and h2)are merged together , and generate hash value h12.

Likewise h3 and h4, h5 and h6, h7 and h8 are merged together separately and generate hash values h34, h56, h78 respectively. Then again h12 and h34, h56 and h78 are merged together separately and generate hash values h1234, h5678 respectively. Then again h1234, h5678 are merged together separately and generate top hash value as h12345678 .This top hash value is given to data owner only and all hash values except this top hash value are stored in cloud server.

Fig. 3: Top Hash Value Generated

D. Trusted Party Auditor

The Trusted Party Auditor is a module which is used to the audit the data that are uploaded by the Data Owner in the Server of the Cloud Service Provider. So that they will audit the data based on the Data Owner's request.Once it received the request from data owner, it checks the data integrity stored in cloud server .In this auditing process [9] first it recieved the original Top Hash Value from data owner. Then it request particular part P1 along with its hash value h1 and also request h2 value from cloud server. Once it received this , it merge this two hash value and generate h12 and request h34 value from cloud server.

Once it received this, it merge this two hash value and generate h1234 and request h5678 value from cloud server. Once it received this, it merge this two hash value and generate h12345678 value which is the new top hash value.Already TPA have original top hash value and now check this new top hash value with original top hash value , if both values are same means then it sends message to data owner that the data is in correct, consistent manner .If both values are different means then it sends message to data owner that the some data are lost from the original information.

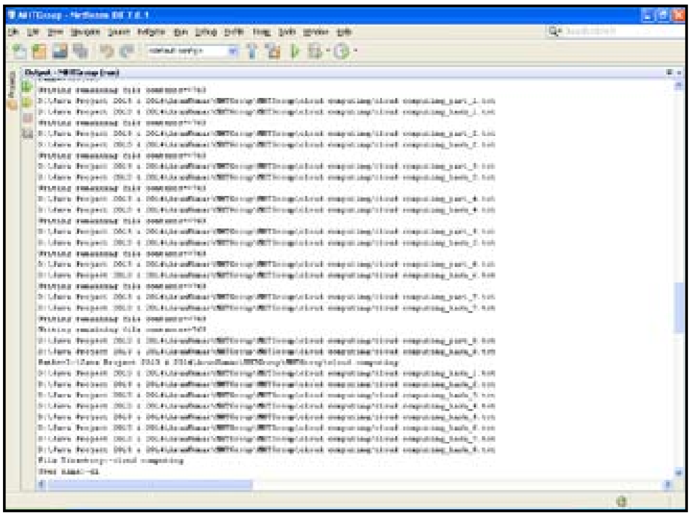


Fig. 4: TPA Audit The Files

E. Random Auditing

In this module we can audit the Data [14] that are not requested by the data owner . So that we can increase security level by identifying any mistake that happens for the data that are stored in the Cloud Server. The Data Owner have to provide their Private key while updating their data in the Cloud Server.

IX. Conclusion

The process such as the data owner can check the integrity of their data stored in cloud server using TPA which can be done in efficient manner.If any modifications find out by the TPA, TPA will immediately intimate to the owner of the file and so security and data integrity is secured properly.TPA would not learn any knowledge about the data content stored on the cloud server during the efficient auditing process, which 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. Cloud data security is an important aspect for the client while using cloud services. Third Party Auditor can be used to ensure the security and integrity of data. Third party auditor can be a trusted third party to resolve the conflicts between the cloud service provider and the client.

References

[1] *Cong Wang, Sherman S.M.Chow, Qian Wang, Kui Ren, and Wenjing Lou, "Privacy Preserving Public Auditing for Secure Cloud Storage", IEEE , Vol.62 , No. 2,February 2013.*

[2] *C.Wang, Q.Wang, K.Ren, and W.Lou, "Privacy Preserving Public Auditing for Storage Security in Cloud Computing", IEEE INFOCOM'10, March 2010.*

[3] *A.Juels and J.Burton, S.Kaliski, "PORs: Proof Of Retrieviability for Large Files", Proc. ACM Conf. Computer and Comm. Security(CCS'07), pp.584-597, October 2007.*

[4] *Dr.Sunitha Abburu, Saranya Eswaran, "Identifying Data Integrity in the Cloud Storage", IJCSI, Vol.9, Issue 2,No. 1, March 2012.*

[5] *Prof.R.Dheenadayalu, M.Sowparnika, "Improving Data Integrity on Cloud Storage Services", IJESI, Vol.2,Issue 2, February 2013.*

[6] *Qian Wang and Cong Wang and Kui Ren, Wenjing Lou, Jin Li "Enabling Public Auditability And Data Dynamics For Storage Security in Cloud Computing" in IEEE transactions on parallel and distributed systems, 2011, vol. 22, no. 5.*

[7] *Farzad Sabahi,"Cloud Computing Security Threats and Responses" ,IEEE confer. 2011, 978-1-61284-486-2/111*

[8] *Ravi Kant Sahu and Abhishek Mohta, L.K. Awasthi "Robust Data Integration While Using Third Party Auditor For Cloud Data Storage Services", conf. IJARCSSE, 2012, Volume 2, Issue 2,ISSN: 2277 128X.*

[9] *Govinda V, and Gurunathaprasad, H Sathshkumar,"Third Party Auditing For Security Data Storage in cloud through digital signature using RSA" IJASATR, 2012, issue 2,vol-4, Issn 2249-9954.*

[10] *P. Mell and t. Grance "Draft Nist Working Definition of Cloud Computing", 2009. [11] Elinor Mills, "Cloud Computing Security Forecast: Clear Skies", 2009.*

[11] *Paul Zimski, "Cloud Computing faces Security Storm",in 2009.*

[12] *Jia xu and Ee-chien chang, "Towards efficient proofs of Retrievibility in Cloud Storage".*

[13] *G.Ateniese,R.Burns,R.Curtmola,J.Herring,L.Kissner, Z.Peterson, and D.Song, "Provable Data Possession at untrusted stores", in CCS'07.*

[14] *Dalia Attas and Omar Batrafi, "Efficient Integrity Checking Technique for Securing Client Data in Cloud Computing", in IJECS – IJENS, 2011.*

[15] *R.Sravan Kumar and Saxena, "Data Integrity proofs in Cloud Storage", in IEEE 2011*

K.Meenakshi, Academic carrier starts with Bachelor of Engineering B.E., from Raja College Of Engineering and Technology, and now doing Final Year M.Tech - CSE (Part Time) in Dr.MGR Educational and Research Institute. Having 8 Years of experience from KLNCE, PTRCET, VCE, Madurai , presently working as a Assisstant Professor in Dr.MGR Educational and Research Institute.