Vandana Publications
IJEMR

Volume-7, Issue-2, March-April 2017
International Journal of Engineering and Management Research
Page Number: 80-84

Privacy Preserving Public Auditing of Shared Data in Cloud (PPPA)

Aswathy K S[1], Karthika N[2], Neeraja M R[3], Sneha K S[4], Kavitha S[5]
[1,2,3,4] B.Tech Students, Department of Computer Science and Engineering, Ammini College of Engineering Palakkad, Kerala, INDIA
[5]Assistant Professor,Department of Computer Science and Engineering, Ammini College of Engineering Palakkad, Kerala, INDIA

ABSTRACT

With cloud storage services, it is commonplace for data to be not only stored in the cloud, but also shared across multiple users. However, public auditing for such shared data— while preserving identity privacy — remains to be an open challenge. In our project, we propose the first privacy-preserving mechanism that allows public auditing on shared data stored in the cloud. In particular, we exploit ring signatures to compute the verification information needed to audit the integrity of shared data. Our mechanism, the identity of the signer on each block in shared data is kept private from a third party auditor (TPA), who is still able to verify the integrity of shared data without retrieving the entire file. Our experimental results demonstrate the effectiveness and efficiency of our proposed mechanism when auditing shared data.

Keywords-PublicAuditing, Privacy Preservation, Homomorphic Authenticators

I. INTRODUCTION

Cloud service providers manage an enterprise-class infrastructure that offers a scalable, secure and reliable environment for users, at a much lower marginal cost due to the sharing nature of resources. It is routine for users to use cloud storage services to share data with others in a team, as data sharing becomes a standard feature in most cloud storage offerings, including Drop box and Google Docs. The integrity of data in cloud storage, however, is subject to scepticism and scrutiny, as data stored in an untrusted cloud can easily be lost or corrupted, due to hardware failures and human errors. To protect the integrity of cloud data, it is best to perform public auditing by introducing a third party auditor (TPA), who offers its auditing service with more powerful computation and communication abilities than regular users.

We believe that sharing data among multiple users is perhaps one of the most engaging features that motivates cloud storage. A unique problem introduced during the process of public auditing for shared data in the cloud is how to preserve identity privacy from the TPA, because the identities of signers on shared data may indicate that a particular user in the group or a special block in shared data is a higher valuable target than others. For example, Alice and Bob work together as a group and share a file in the cloud. The shared file is divided into a number of small blocks, which are independently signed by users. Once a block in this shared file is modified by a user, this user needs to sign the new block using her public/private key pair. The TPA needs to know the identity of the signer on each block in this shared file, so that it is able to audit the integrity of the whole file based on requests fromAlice or Bob.

II. GOAL OF THE PROJECT

We propose a new privacy preserving public auditing mechanism for shared data in an untrusted cloud.We utilize ring signatures to construct homomorphic authenticators, so that the third party auditor is able to verify the integrity of shared data for a group of users without retrieving the entire data — while the identity of the signer on each block in shared data is kept private from the TPA. In addition, we further extend our mechanism to support batch auditing, which can audit multiple shared data simultaneously in a single auditing task. Meanwhile, PPPA continues to use random masking to support data privacy during public auditing, and leverage index hash tables to support fully dynamic operations on shared data. A dynamic operation indicates an insert, delete or update operation on a single block in shared data. A high-level comparison between proposed system and existing mechanisms in the literature is shown. To our best knowledge, this represents the first attempt towards designing an effective privacy preserving public auditing mechanism for shared data in the cloud

III. EXISTING SYSTEM

Many mechanisms have been proposed to allow not only a data owner itself but also a public verifier to efficiently perform integrity checking without downloading the entire data from the cloud, which is referred to as public auditing . In these mechanisms, data is divided into many small blocks, where each block is independently signed by the owner; and a random combination of all the blocks instead of the whole data is retrieved during integrity checking . A public verifier could be a data user (e.g., researcher) who would like to utilize the owner's data via the cloud or a third-party auditor (TPA) who can provide expert integrity checking services.

Moving a step forward, Wang et al. designed an advanced auditing mechanism .so that during public auditing on cloud data, the content of private data belonging to a personal user is not disclosed to any public verifiers. Unfortunately, current public auditing solutions mentioned above only focus on personal data in the cloud .We believe that sharing data among multiple users is perhaps one of the most engaging features that motivates cloud storage. Therefore, it is also necessary to ensure the integrity of shared data in the cloud is correct.

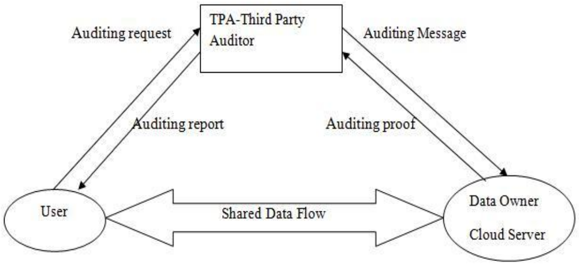
Existing public auditing mechanisms canactually be extended to verify shared data integrity. However, a new significant privacy issue introduced in the case of shared data with the use of existing mechanisms is the leakage of identity privacy to public verifiers. However, a new significant privacy issue introduced in the case of shared data with the use of existing mechanisms is the leakage of identity privacy to public verifiers.

IV. PROPOSED SYSTEM

To solve the above privacy issue on shared data, we propose,a novel privacy-preserving public auditing mechanism. More specifically, we utilize ring signatures to construct homomorphic authenticators, so that a public verifier is able to verify the integrity of shared data without retrieving the entire data while the identity of the signer on each block in shared data is kept private from the public verifier.In addition, we further extend our mechanism to support batch auditing, which can perform multiple auditing tasks simultaneously and improve the efficiency of verification for multiple auditing tasks. Meanwhile, it is compatible with random masking, which has been utilized in WWRL and can preserve data privacy from public verifiers. Moreover, we also leverage index hash tables from a previous public auditing solution to support dynamic data. A high-level comparison among ours and existing mechanisms is presented.

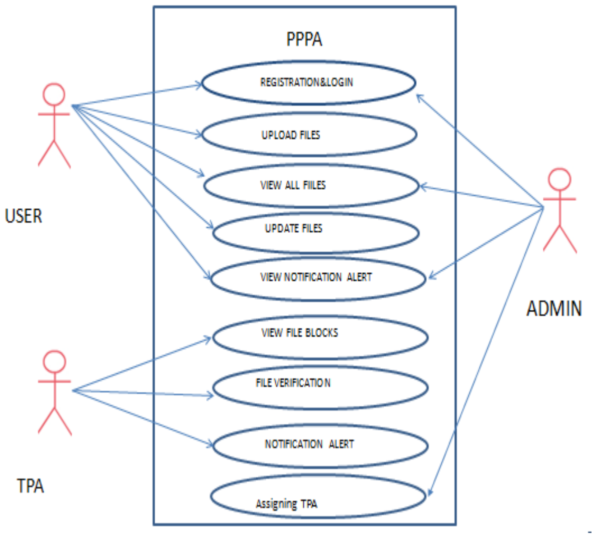
V. SYSTEM DESIGN

The architecture involves three parties: the cloud server, the third party auditor (TPA) and users. There are two types of users in a group: the original user and a number of group users.The original user and group users are both members of the group. Group members are allowed to access and modify shared data created by the original user based on access control polices. Shared data and its verification information (i.e. signatures) are both stored in the cloud server. The third party auditor is able to verify the integrity of shared data in the cloud server on behalf of group members. Our system model includes the cloud server, the third party auditor and users. The user is responsible for deciding who is able to share her data before outsourcing data to the cloud. When a user wishes to check the integrity of shared data, she first sends an auditing request to the TPA. After receiving the auditing request, the TPA generates an auditing message to the cloud server, and retrieves an auditing proof of shared data from the cloud server. Then the TPA verifies the correctness of the auditing proof. Finally, the TPA sends an auditing report to the user based on the result of the verification.



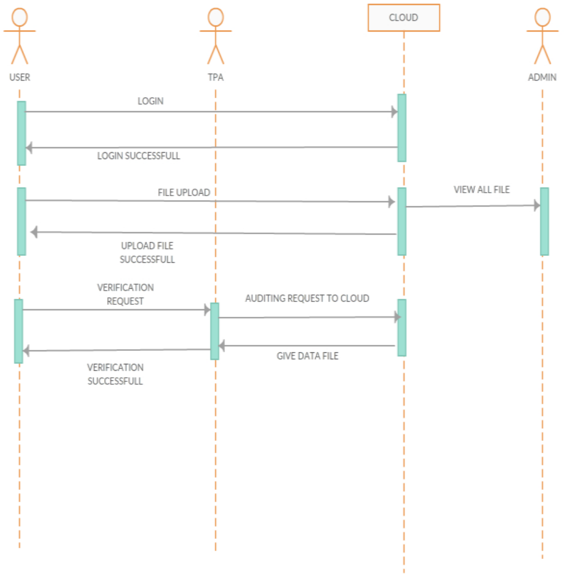
Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.



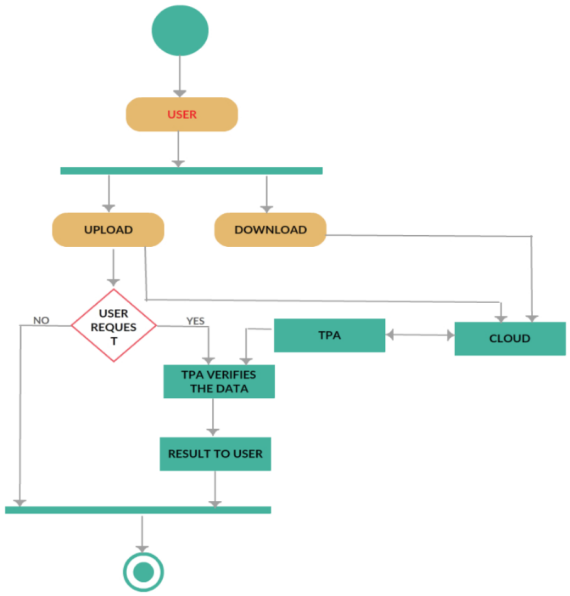
VI. SEQUENCE DIAGRAM

A Sequence diagram is an interaction diagram that shows how processes operate with oneanother and in what order. It is a construct of a Message Sequence Chart. A sequence diagramshows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry outthe functionality of the scenario.



VII. ACTIVITY DIAGRAM

Activity diagrams are graphical representations of workflows of stepwise activities andactions with support for choice, iteration and concurrency. In the Unified Modeling Language,activity diagrams are intended to model both computational and organizational processes (i.e.workflows). Activity diagrams show the overall flow of control.



VIII. DATA FLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of the "flow" of data through aninformation system, modelling its process aspects. A DFD is often used as a preliminary step tocreate an overview of the system, which can later be elaborated. DFDs can also be used for thevisualization of data processing

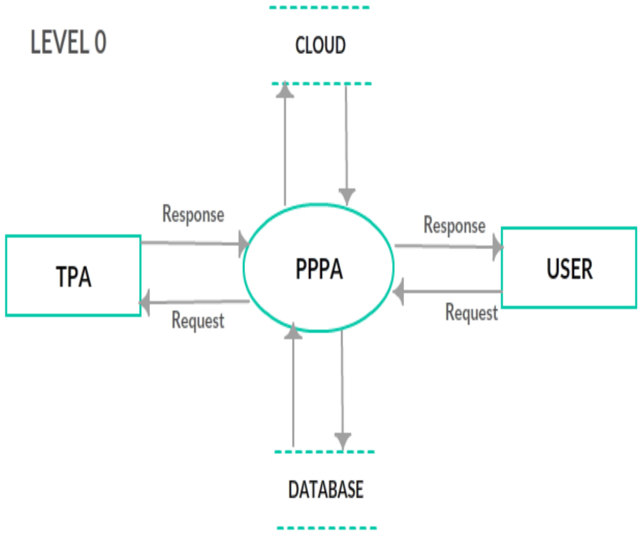


Figure:level 0

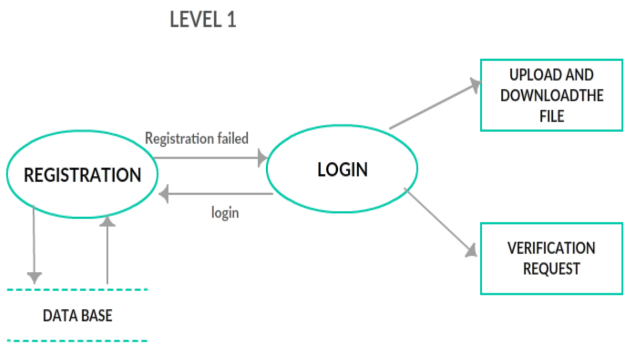


Figure : level 1user

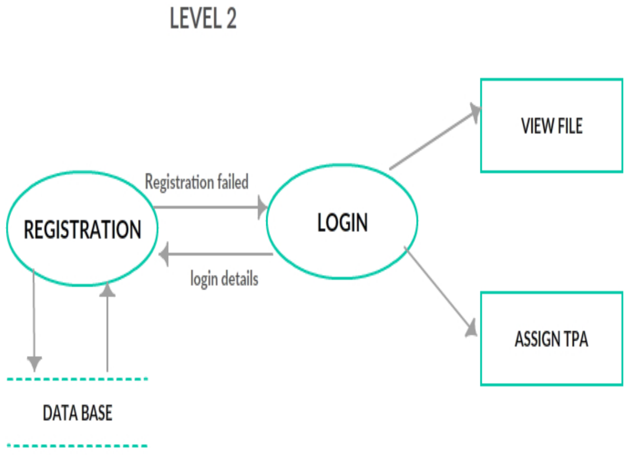


Figure:level 2 admin

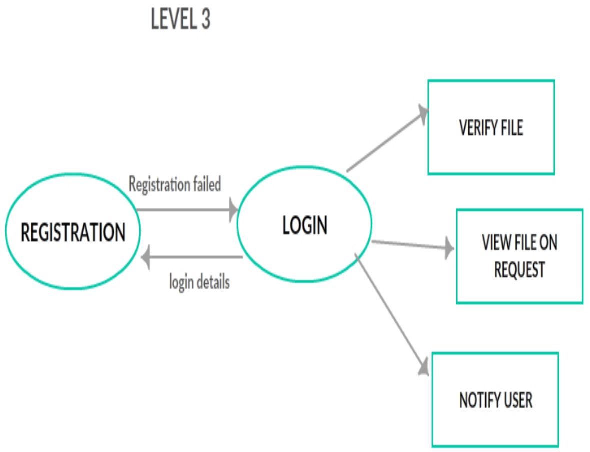


Figure: level 3 TPA

IX. MODULE DESIGN

- **OWNER LOGIN & REGISTRATION:**

In this module an owner has to upload its files in a cloud server, he/she should register first. Then only he/she can be able to do it. For that he needs to fill the details in the registration form. These details are maintained in a database. Any of the above mentioned person have to login, they should login by giving their email id and password.

- **USER REGISTRATION & LOGIN:**

In this module if a user wants to access the data which is stored in a cloud,he/she should register their details first. These details are maintained in a Database. If the user is an authorized user,he/she can download the file by using file id which has been stored by data owner when it was uploading.

- **THIRD PARTY AUDITOR REGISTRATION & LOGIN:**

In this module, if a third party auditor TPA(maintainer of clouds) wants to do some cloud offer , they should register first. Here we are doing like, this system allows only three cloud service providers. After third party auditor gets logged in, He/ She can see how many data owners have uploaded their files into the cloud. Here we are providing three tpa for maintaining three different clouds.

- **VERIFICATION**

The TPA verifies the files when the user sends the auditing request to the TPA inorder to check the integrity of data. The TPA does this with the help of Homomorphic Authenticator ring signatures.

X. CONCLUSION

In this paper, we propose the first privacypreservingpublic auditing mechanism for shared data in the cloud. We utilize ring signatures to construct homomorphic authenticators, so the TPA is able to audit the integrity of shared data, yet cannot distinguish who is the signer on each block, which can achieve identity privacy. To improve the efficiency of verification for multiple auditing tasks, we further extend our mechanism to support batch auditing. An interesting problem in our future work is how to efficiently audit the integrity of shared data with dynamic groups while still preserving the identity of the signer on each block from the third party auditor.

REFERENCES

[1] Provable data possession at untrusted stores, G. Ateniese, R. Burns, ACM Conference on Computer and Communications Security, February 2007.

[2] Privacy preserving Public Auditing for Data Storage Security in Cloud Computing, C.Wang, Q. Wang, International Conference on Computer Communications, 2010, pp. 525–533.

[3] How to Leak a Secret, R. L. Rivest, A. Shamir, and Y. Tauman, International conference on the theory and application of cryptology and information security, 2001, pp. 552–565.
[4] Scalable and efficient provable data possession, G. Ateniese, R. D. Pietro, L. V. Mancini, and G. Tsudik, International Conference on Security and Privacy in CommunicationNetworks, 2008.
[5] Ensuring data storage Security in Cloud Computing, C. Wang, Q. Wang, K. Ren, and W. Lou, International workshop on quality of service, feb 2009