INTERNATIONAL JOURNAL OF
INNOVATIVE RESEARCH & DEVELOPMENT

ISSN: 2278 – 0211

Enabling Public Auditability And Data Dynamics For Storage Security In Cloud Computing

M.Mrinalni Vaknishadh
Centre for Information Technology and Engineering
Manonmaniam Sundaranar University
Tamil Nadu, India

Abstract:

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 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.

Introduction

This projects deal with Several trends are opening up the era of Cloud Computing, which is an Internet-based development and use of computer technology.Cloud brings about many challenging design issues which have profound influence on the security and performance of the overall system. One of the biggest concerns with cloud data storage is that of data integrity verification at untrusted servers

In order to solve the problem of data integrity checking, many schemes are proposed under different systems and security models.Considering the role of the verifier in the model, all the schemes presented before fall into two categories: private auditability and public auditability. clients are able to delegate the evaluation of the service performance to an independent third party auditor (TPA), without devotion of their computation resources

Existing System

Although the existing schemes aim at providing integrity verification for different data storage systems, the problem of supporting both public auditability and data dynamics has not been fully addressed.

Proposed System

Client

an entity, which has large data files to be stored in the cloud and relies on the cloud for data maintenance and computation.

Cloud Storage Server (CSS)

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

Third Party Auditor (TPA)

an entity, which has expertise and capabilities that clients do not have, is to assess and expose risk of cloud storage services on behalf of the clients upon request.

Module Description

Organization and ID creation

In this module the organization has to register for getting the data service. The org. Head can further give the details about its employee and create their ID's.

Data Dynamics Operations

It has data to be stored in the cloud and rely on the cloud for data computation, consist of both individual consumers and organizations. The data stored in the cloud may be frequently updated by the users, including insertion, deletion, modification, appending, reordering, etc.

Public Auditing

The public auditing system of data storage security in Cloud Computing, and propose a protocol supporting for fully dynamic data operations, especially to support block insertion.

Batch Auditing

As cloud servers may concurrently handle multiple verification Different data's from different clients, given signatures on distinct data files from different clients.

RSA Process

Considering various kinds of data for each user stored in the cloud and the demand of long term continuous assurance of their data safety.

TPA Services

It has significant resources and expertise in building and managing distributed cloud storage servers, owns and operates live Cloud Computing systems.

Testing

Software testing is one element of a broader topic that is verification and validation. Tests are conducted with the intention of finding errors in the software.

The testing stage has several purposes to affirm the quality of the project, to find and eliminate any residual errors from the previous stage to validate the software and to eliminate the operational reliability of the system.

There are two general category of testing. Pre implementation and post implementation.

Testing Process

Software Testing is the execution of program to find its faults. The testing process focuses on the logical internals of the software, ensuring that all the statements been test and on the functional externals, that is conducting tests to uncover errors and ensure that defined inputs will produce actual results that agree with required results. Testing software is a four-folded analysis, viz.

- **Unit testing**
- **Integration testing**
- **User acceptance testing**
- **Validation testing**

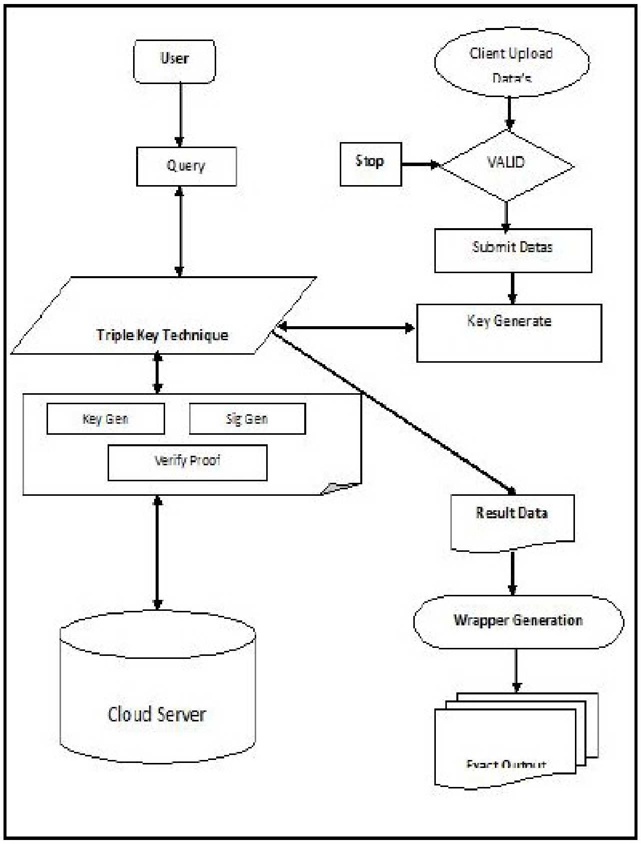

Figure 1 :Architecture Design

Conclusion

We propose a "Ensuring for data storage security in Cloud Computing", where TPA can perform the **storage auditing without demanding the local copy of data.**

We utilize the tripe key cryptography to guarantee that 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.

Considering TPA may on currently handle multiple audit sessions from different users for their outsourced data files, we further extend our privacy-preserving public auditing protocol into a multi-user setting, where TPA can perform the multiple auditing tasks in a batch manner, i.e., simultaneously. Extensive security and performance analysis shows that the proposed schemes are provably secure and highly efficient. We believe all these advantages of the proposed schemes will shed light on economies of scale for Cloud Computing.

Reference

1. **Q. Wang, C. Wang, J. Li, K. Ren, and W**. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing,"
2. G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores,"
3. **A. Juels and B. S. Kal**iski, Jr., "Pors: proofs of retrievability for large files,"
4. H. Shacham and B. Waters, "Compact proofs of retrievability.
5. Q. Wang, K. Ren, W. Lou, and Y. Zhang, "Dependable and secure
6. sensor data storage with dynamic integrity assurance,"
7. **G. Ateniese, R**. D. Pietro, L. V. Mancini, and G. Tsudik, "Scalable and efficient provable data possession,"