International Journal of Research in Engineering, Science and Management
Volume-1, Issue-12, December-2018
www.ijresm.com | ISSN (Online): 2581-5792

Beat Separate Enquiry

R. Nanda Kumar[1], K. Vinodh Kumar[2]
[1]*Assistant Professor, Dept. of Computer Science and Engineering, Thanagavelu Engg. College, Chennai, India*
[2]*Research Scholar, Department of Information Technology, St. Peter's University, Chennai, India*

***Abstract*: In this paper, we propose a secure certificate less public integrity verification scheme (SCLPV). The SCLPV is the first work that simultaneously supports certificate less public verification and resistance against malicious auditors to verify the integrity of outsourced data in CPSS. A formal security proof proves the correctness and security of our scheme. In addition, an elaborate performance analysis demonstrates that the SCLPV is efficient and practical. Compared with the only existing certificate less public verification scheme (CLPV), the SCLPV provides stronger security guarantees in in terms of remedying the security vulnerability of the CLPV and resistance against malicious auditors. In comparison with the best of integrity verification scheme achieving resistance against malicious auditors, the communication cost between the auditor and the cloud server of the SCLPV is independent of the size of the processed data; meanwhile, the auditor in the SCLPV does not need to manage certificates.**

***Keywords*: CPSS secure certificate less public integrity verification scheme (SCLPV) regenerating-code**

1. Introduction

Cyber-physical-social system (CPSS) has been envisioned as the next phase of computing systems. It combines measured elements of the physical world with manual human input, and seamlessly integrates physical components with traditional social networks. Typically, CPSS allows users to store and share information, locations, and trajectories collected from personal devices, such as smart phones and sensors. These extra data from physical world, which are numerously generated by CPSS users and collected by enterprises each day, are extraordinarily valuable not only to individuals themselves but also to enterprises to better understand people's daily activities, social areas, and life patterns . From data owners‟ perspective, including both individuals and enterprises, outsourcing their data to cloud servers is a wise and practical choice, because cloud servers provide users an efficient and flexible service to manage data.

A. Scope of the project

We propose a secure certificate less public integrity verification scheme (SCLPV) against malicious auditors for cloud storage in CPSS. In the SCLPV, a public auditor is able to verify the integrity of outsourced data without retrieving the entire data set and managing the user's certificate. Meanwhile, to fight against malicious auditors, the SCLPV requires CPSS users periodically check their auditors‟ behaviors. Furthermore, we extensively analyze the performance of the SCLPV and demonstrate that the SCLPV is efficient and practical.

B. Problem statement

The large size of the outsourced data and the user's constrained resource capability, the tasks of auditing and reparation in the cloud can be formidable and expensive.

C. Existing system

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. The large size of the outsourced data and the user's constrained resource capability, the tasks of auditing and reparation in the cloud can be formidable and expensive for the users.

D. Drawbacks in existing system

Absence of data owner is not possible, Reparation problem of failed authenticators

2. Proposed system

In our proposed system Public Auditability which allow TPA to verify the intactness of the data in the cloud on demand without introducing additional online burden to the data owner. Our scheme is the first to allow privacy-preserving public auditing for regenerating code- based cloud storage. The coefficients are masked by a PRF (Pseudorandom Function) during the Setup phase to avoid leakage of the original data.

A. Advantages in proposed system

Absence of data owner is possible, Reparation problem of authenticators is solved.

B. Literature survey

H. Chen and P. Lee, "Enabling data integrity protection in regenerating coding-based cloud storage: Theory and implementation," To protect outsourced data in cloud storage against corruptions, enabling integrity protection, fault tolerance, and efficient recovery for cloud storage becomes critical. Regenerating codes provide fault tolerance by striping data across multiple servers, while using less repair traffic than traditional erasure codes during failure recovery. Therefore, we study the problem of remotely checking the integrity of regenerating-coded data against corruptions under a real-life

cloud storage setting. We design and implement a practical data integrity protection (DIP) scheme for a specific regenerating code, while preserving the intrinsic properties of fault tolerance and repair traffic saving. Our DIP scheme is designed under a Byzantine adversarial model, and enables a client to feasibly verify the integrity of random subsets of outsourced data against general or malicious corruptions. It works under the simple assumption of thin-cloud storage and allows different parameters to be fine-tuned for the performance-security trade-off. We implement and evaluate the overhead of our DIP scheme in real cloud storage test be under different parameter choices. We demonstrate that remote integrity checking can be feasibly integrated into regenerating codes in practical deployment. Michael Armbrust, Armando Fox," Above the Clouds: A View of Cloud Computing", IEEE Transactions on, vol. 24, no. 9, pp. 1717–1726, 2013: Cloud Computing, the long-held dream of computing as a utility, has the potential to transform a large part of the IT industry, making software even more attractive as a service and shaping the way IT hardware is designed and purchased. Developers with innovative ideas for new Internet services no longer require the large capital outlays in hardware to deploy their service or the human expense to operate it. They need not be concerned about over-provisioning for a service whose popularity does not meet their predictions, thus wasting costly resources, or under-provisioning for one that becomes wildly popular, thus missing potential customers and revenue. Moreover, companies with large batch-oriented tasks can get results as quickly as their programs can scale, since using 1000 servers for one hour costs no more than using one server for 1000 hours. This elasticity of resources, without paying a premium for large scale, is unprecedented in the history of IT.

3. Project description

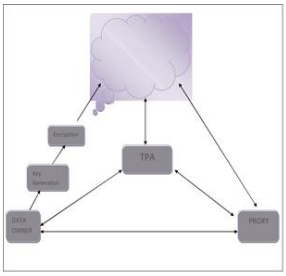

Fig. 1. System architecture

Cloud computing is defined as a type of computing that relies on sharing computing resources rather than having local servers or personal devices to handle applications. Cloud storage services have rapidly become increasingly popular. Users can store their data on the cloud and access their data anywhere at any time. It is noted that data owners lose ultimate control over the fate of their outsourced data; thus, the correctness, availability and integrity of the data are being put at risk. On the one hand, the cloud service is usually faced with a broad range of internal/external adversaries, who would maliciously delete or corrupt users‟ data.

A. Methodologies: data owner interface design

This is the first module of our project. The important role for the data owner is to move login window to data owner window. This module has created for the security purpose. In this login page we have to enter login user id and password. It will check username and password is match or not (valid user id and valid password). If we enter any invalid username or password we can't enter into login window to data owner window it will shows error message. So we are preventing from unauthorized data owner entering into the login window to data owner window. It will provide a good security for our project. So server contain user id and password server also check the authentication of the user. It well improves the security and preventing from unauthorized data owner enters into the network. In our project we are using JSP for creating design. Here we validate the login user and server authentication.

B. Key generation & files upload

This Module is Key Generation and File Upload is used to Data Owner is Generate Public key and Secret Key. The Data Owner is sent to Secret key in Proxy. Then Data Owner is Upload the text File in Cloud. Before upload the Text File due to Encryption Process done and Upload the file in Cloud.

C. Third party auditor checking

This Module is Third Party Auditor Checking that is Public Auditing Process. The Third Party Auditor Always check in the Cloud Data Owner Files Corrupt or not. Sometimes search to Data owner files in the cloud. Suppose The Data Owner Files are corrupted .Third Party Auditor sent Audit result to proxy.

D. File corruption

This Module is File Corruption. The user used in Cloud Stored Files. User download the File and to be used. Suppose User Edit the File at the same time File to be corrupted. The Alert message automatically sent to the Third Party Auditor. The Third Party Auditor receives and forwards the Alert message to Proxy.

E. Integrity checking and regeneration

This Module is integrity checking and regeneration. The Third Party Auditor Always check in the Cloud Data Owner Files Corrupt or not. Suppose The Data Owner Files are corrupted .Third Party Auditor sent Audit result to proxy. The Proxy is sent request to cloud in particular Corrupted

4. Conclusion

In this paper, we propose a public auditing scheme for their generating-code-based cloud storage system, where the data owners are privileged to delegate TPA for their data validity checking. To protect the original data privacy against the TPA, we randomize the coefficients in the beginning rather than applying the blind technique during the auditing process. Considering that the data owner cannot always stay online

imprecise, in order to keep the storage available and verifiable after a malicious corruption, we introduce a semi-trusted proxy into the system model and provide a privilege for the proxy to handle the reparation of the coded blocks and authenticators .To better appropriate for the regenerating-code-scenario, we design our authenticator based on the BLS signature. This authenticator can be efficiently generated by the data owner simultaneously with the encoding procedure. Extensive analysis shows that our scheme is provable secure, and the performance valuation shows that our scheme is highly efficient and can be feasibly integrated into a regenerating-code-based cloud storage system

5. Future enhancements

In Future we can implement this protocol using multi cloud or multi server. Code implementation eliminates the encoding requirement of storage nodes (or cloud) during repair, while ensuring that the new set of stored chunks after each round of repair preserves the required fault tolerance.

References

[1] Yuan Zhang,Chuxiang Xu,Shui Yu,Hongwei Li and Xiaojun Zhang ,"Secure Certificateless Public Verification For Cloud-Based Cyber-Physical-Social Systems against Malicious Auditors", IEEE Transactions on computational social system, vol. 2, no. 4, Feb 2016.

[2] H. Chen and P. Lee, "Enabling data integrity protection in regenerating coding-based cloud storage: Theory and implementation," Parallel and Distributed Systems, IEEE Transactions on, vol. 25, no. 2, pp. 407–416, Feb 2014.

[3] Michael Armbrust, and Armando Fox, "Above the Clouds: A View of Cloud Computing", IEEE Transactions on, vol. 24, no. 9, pp. 1717–1726, 2013.

[4] Y. Zhu, H. Hu, G. J. Ahn, and M. Yu, "Cooperative provable data possession for integrity verification in multi cloud storage," Parallel and Distributed Systems, IEEE Transactions on, vol. 23, no. 12, pp. 2231–2244, 2012.