Public Auditing for Shared Data using Cloud

Miss. Ghadage Kavita B.[1] Miss. Netake Aarti T.[2] Prof. Mrs.Archana H. Renushe[3]
[1,2]BE Student [3]Assistant Professor
[1,2,3]Department of Computer Science & Engineering
[1,2,3]Dr.Daulatrao Aher College of Engg, Karad, India

Abstract— The data services in the cloud, users can easily modify and share data as a group. To ensure data integrity can be audited publicly, users need to compute signatures on all the blocks in shared data. Different blocks are signed by different users key due to data modifications performed by different users. For security reasons, once a user is revoked from the group, it cannot access the data in the group. The straightforward method, which allows an existing user to download the corresponding part of shared data and re-sign it during user revocation, is inefficient due to the large size of shared data in the cloud. In this paper, we propose a novel public auditing mechanism for the integrity of shared data with efficient user revocation in the cloud. By utilizing proxy re-signatures, we allow the cloud to re-sign blocks on behalf of existing users during user revocation, so that existing users do not need to download and re-sign blocks by themselves. In addition, a public verifier is always able to audit the integrity of shared data without retrieving the entire data from the cloud, even if some part of shared data has been re-signed by the cloud. Experimental results show that our mechanism can significantly improve the efficiency of user revocation.

Keywords: Public Auditing Shared Data, RSA-PSS algorithm, Cloud

I. Introduction

Based on the new proxy re-signature scheme and its properties in the existing System, we now present Public Auditing Shared Data using ECSDA and RSA-PKCS Algorithm. In our project, the original user acts as the group manager, who is able to revoke users from the group when it is necessary. Meanwhile, we allow the cloud to perform as the semi-trusted proxy and translate signatures for users in the group with resigning keys. As emphasized in recent work, for security reasons, it is necessary for the cloud service providers to storage data and keys separately on different servers inside the cloud in practice. Therefore, we assume the cloud has a server to store shared data, and has another server to manage resigning keys. To ensure the privacy of cloud shared data at the same time, additional mechanisms, such as, can be utilized. The main focus of this project is to audit the integrity of cloud shared data.

II. Literature Review

A. [1].“Public Auditing for Shared Data with Efficient User Revocation in the Cloud,” B. Wang,Li, and H. Li, in the Proceedings of IEEE INFOCOM 2013, 2013, pp. 2904–2912.

Techniques used in Public Auditing on Cloud There are some different techniques which used in different auditing mechanisms. This section introduce some the techniques like MAC, HLA etc. which are used for different purposes like data authentication, data integrity in auditing schemes on cloud.

B. [2].”Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing”C.Wang, Q. Wang, K. Ren, and W.lou, in the Proceedings of IEEE INFOCOM 2010, 2010, pp. 525–533.

With cloud data services, it is possible to all or common place for data to be not on stored in the cloud, but also shared across multiple users. Unfortunately, the integrity of cloud data is subject to misconception due to the existence of hardware/software failures and human errors. To allow both data owners and public verifiers several mechanisms have been designed for efficiently auditing cloud data integrity without retrieving the entire data from the cloud server. However, public auditing on the integrity of shared data with these previously existing mechanisms will inevitably reveal confidential information, identity & privacy to public verifiers. In this work a novel privacy-preserving mechanism used to supports public auditing on shared data stored in the cloud. In particular, here exploit ring signatures is used which computes verification of metadata on user demand and audit the correctness of shared data.

C. [3].“Provable Data Possession at Untrusted Stores,”G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, in the Proceedings of ACM CCS 2007, 2007, pp. 598–610.

In this model the client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it. The model generates probabilistic proofs of possession by sampling random sets of blocks from the server, which drastically reduces I/O costs. The client maintains a constant amount of metadata to verify the proof. The challenge/response protocol transmits a small, constant amount of data, which minimizes network communication. Thus, the PDP model for remote data checking supports large data sets in widely-distributed storage systems.

We present two provably-secure PDP schemes that are more efficient than previous solutions, even when compared with schemes that achieve weaker guarantees. In particular, the overhead at the server is low (or even constant), as opposed to linear in the size of the data. Experiments using our implementation verify the practicality of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation.

D. [4]. “Proxy Re-signatures: New Definitions, Algorithms and Applications,”G. Ateniese and S. Hohenberger, in the Proceedings of ACM CCS 2005, 2005, pp. 310–319.

In a proxy re-signature scheme, a semi-trusted proxy is given some information which allows it to transform Alice’s signature on a message m into Bob's signature on m, but the proxy cannot, on its own, generate signatures for either Alice or Bob.

Proxy signatures [18] allow Alice to delegate her signing rights to Bob but only if the proxy cooperates. In practice, Bob and the proxy can jointly generate a signature on arbitrary messages on Alice's behalf. This is usually accomplished by dividing Alice's secret into two shares which are distributed to Bob and the proxy (each gets only one share). A signature from Alice on a message is generated by combining two partial signatures on the same message computed by Bob and the proxy under their own shares, respectively.

We begin our results by formalizing the definition of security for a proxy re-signature. We next substantiate the need for improved schemes by pointing out certain weaknesses of the original BBS proxy re-signature scheme which make it unfit for most practical applications.

We then present two secure proxy re-signature schemes based on bilinear maps. Our first scheme relies on the Computational Diffie-Hellman (CDH) assumption; here the proxy can translate from Alice to Bob and vice-versa. Our second scheme relies on the CDH and 2-Discrete Logarithm (2-DL) assumptions and achieves a stronger security guarantee – the proxy is only able to translate in one direction. Constructing such a scheme has been an open problem since proposed by BBS in 1998. Furthermore in this second scheme, even if the delegator and the proxy collude, they cannot sign on behalf of the delegate. Both schemes are efficient and secure in the random oracle model.

E. [5]. "Towards Secure andDependable Storage Services in Cloud Computing," C. Wang, Q. Wang, K. Ren, and W. Lou, IEEE Transactions on Services Computing, vol. 5, no. 2, pp. 220–232, 2011.

Cloud storage enables users to remotely store their data and enjoy the on-demand high quality cloud applications without the burden of local hardware and software management. Though the benefits are clear, such a service is also relinquishing users' physical possession of their outsourced data, which inevitably poses new security risks towards the correctness of the data in cloud.

In order to address this new problem and further achieve a secure and dependable cloud storage service, we propose in this paper a flexible distributed storage integrity auditing mechanism, utilizing the homomorphic token and distributed erasure-coded data. The proposed design allows users to audit the cloud storage with very lightweight communication and computation cost. The auditing result not only ensures strong cloud storage correctness guarantee, but also simultaneously achieves fast data error localization, i.e., the identification of misbehaving server. Considering the cloud data are dynamic in nature, the proposed design further supports secure and efficient dynamic operations on outsourced data, including block modification, deletion, and append. Analysis shows the proposed scheme is highly efficient and resilient against Byzantine failure, malicious data modification attack, and even server colluding attacks.

III. CONCLUSION

In this project, we proposed a novel public auditing mechanism for the integrity of shared data with efficient user revocation using two secure algorithm ECDSA and RSA-PSS algorithm. To protect the integrity of shared data, each block in shared data is attached with a signature, once a user modifies a block, he/she must resign the modified block. When a user in the group is revoked, we allow the semi-trusted cloud to re-sign blocks that were signed by the revoked user with proxy re-signatures.

Cloud can improve the efficiency of user revocation so that existing users in the group can save a significant amount of computation and communication resources during user revocation.

REFERENCES

[1] B. Wang, B. Li, and H. Li, "Public Auditing for Shared Data with Efficient User Revoation in the Cloud," in the Proceedings of IEEE INFOCOM 2013, 2013, pp. 2904–2912.

[2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud omputing," Communications of theACM, vol. 53, no. 4, pp. 50–58, Apirl 2010.

[3] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores,"in the Proceedings of ACM CCS 2007, 2007, pp. 598–610.

[4] H. Shacham and B. Waters, "Compact Proofs of Retrievability,"in the Proceedings of ASIACRYPT 2008. Springer-Verlag, 2008, pp.90–107.

[5] C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring Data Storage Security in Cloud Computing," in the Proceedings of ACM/IEEE IWQoS 2009, 2009, pp. 1–9.

[6] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling Public Verifiability and Data Dynamic for Storage Security in Cloud Computing," in the Proceedings of ESORICS 2009. Springer-Verlag,2009, pp. 355–370.