ISSN: 2321-7782 (Online)
Volume 3, Issue 1, January 2015
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Privacy-Retaining Public Analyzing for Shared Data in the Cloud

Madhubala Rajendra Patil[1]
Department of Computer Engineering
G. S. Moze College of Engineering,
Savitribai Phule Pune University,
Balewadi, Pune-411045, India

Srinu Dharavath[2]
Department of Computer Engineering
G. S. Moze College of Engineering,
Savitribai Phule Pune University,
Balewadi, Pune-411045, India

Abstract: ***In Cloud Storage, users can remotely store their data and enjoy the on-demand high quality applications and services. The integrity of cloud data is subject to skepticism due to the existence of hardware/software failures and human errors. Several mechanisms have been designed to allow both data owners and public verifiers to efficiently audit cloud data integrity without retrieving the entire data from the cloud server. However, public analyzing on the integrity of shared data with these existing mechanisms will supports public analyzing on shared data stored in the cloud that exploit ring signature to compute verification metadata needed to audit the correctness of shared data.so that a third party auditor (TPA) is able to verify the integrity of shared data for users without retrieving the entire data. Meanwhile, the identity of the signer on each block in shared data is kept private from the TPA also able to perform multiple auditing tasks simultaneously instead of verifying them one by one .In this paper, we proved the data freshness(proved the cloud possesses the latest version of shared data)while still retaining identity privacy. Our experimental result ensures that retrieved data always reflects the most recent updates and prevents rollback attacks.***

Keywords: ***AFS (Authenticated File System); data freshness; public auditing; shared data***

I. INTRODUCTION

With cloud computing and storage, users are able to access and to share resources offered by cloud service providers at a lower marginal cost. It is routine for users to leverage cloud storage services to share data with others in a group, as data sharing becomes standard feature in most cloud storage offerings, including Drop box, iCloud and Google Drive . The integrity of data in cloud storage, however, is subject to skepticism and scrutiny, as data stored in the cloud can easily be lost or corrupted due to the inevitable hardware/software failures and human errors , The traditional approach for checking data correctness is to retrieve the entire data from the cloud, and then verify data integrity by checking the correctness of signatures (e.g., RSA) or hash values (e.g., MD5)of the entire data. Certainly, this conventional approach able to successfully check the correctness of cloud data. However, the efficiency of using this traditional approach on cloud data is in doubt. The main reason is that the size of cloud data is large in general. Downloading the entire cloud data to verify data integrity will cost or even waste users amounts of computation and communication resources, especially when data have been corrupted in the cloud.

Recently, 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. Existing public auditing mechanisms can actually be extended to verify shared data integrity and data freshness.

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 .To protect the confidential information, it is essential and critical to preserve identity privacy from public verifiers during public auditing. To solve the above privacy issue on shared data, we propose Oruta1, a novel privacy retaining public auditing mechanism. More specifically, we utilize ring signatures to construct holomorphic authenticators in Oruta, 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, extend this mechanism to support batch auditing, which can perform multiple auditing tasks simultaneously and improve the efficiency of verification for multiple auditing tasks. Meanwhile, Oruta is compatible with random masking. Oruta stands for "One Ring to Rule Them All".

II. New Ring Signature Scheme

1. *Overview:*

The design of new homomorphic authenticable ring signature (HARS) scheme, which is extended from a classic ring signature scheme [15]. The ring signatures generated by HARS are not only able to preserve identity privacy but also able to support block less verifiability. We will show how to build the privacy retaining public auditing mechanism for shared data in the cloud based on this new ring signature scheme in the next section

2. *Construction of HARS*

HARS contains three algorithms: **Key Gen**, **Ring Sign** and **Ring Verify**. In **Key Gen**, each user in the group generates his/her public key and private key. In **Ring Sign**, a user in the group is able to generate a signature on a block and its block identifier with his/her private key and all the group members' public keys. A block identifiers a string that can distinguish the corresponding block from others. A verifier is able to check whether a given block is signed by a group member in **Ring Verify**.

3. *Security Analysis of HARS*

We discuss security properties of HARS, including correctness, unforgeability, block less verifiability, non-malleability and identity privacy

III. Public Auditing Mechanism

1. *Overview:*

Using HARS and its properties, we now construct Oruta, a privacy retaining public auditing mechanism for shared data in the cloud. With Oruta, the public verifier can verify the integrity of shared data without retrieving the entire data. Meanwhile, the identity of the signer on each block in shared data is kept private from the public verifier during the auditing

2. *Construction of Oruta*

Now, we present the details of our public auditing mechanism. It includes five algorithms: KeyGen, SigGen, Modify, Proof Gen and Proof Verify. In Key-Gen, users generate their own public/private key pairs .In SigGen, a user (either the original user or a group user) is able to compute ring signatures on blocks in shared data by using its own private key and all the group members' public keys. Each user in the group is able to perform an insert, delete or update operation on a block, and compute the new ring signature on this new block in Modify. ProofGen is operated by a public verifier and the cloud server together to interactively generate a proof of possession of shared data. In Proof Verify, the public verifier audits the integrity of shared data by verifying the proof. Note that for the ease of understanding, we first assume the group is static, which means the group is predefined before shared data is created in the cloud and the membership of the group is not changed during data sharing. Specifically, before the original user outsources shared data to the cloud, he/she decides all the group members. Dynamic

Groups: We now discuss the scenario of dynamic groups under our proposed mechanism. If anew user can be added in the group or an existing user can be revoked from the group, then this groups denoted as a dynamic group. To support dynamic groups while still allowing the public verifier to perform public auditing, all the ring signatures on shared data need to be re-computed with the signer's private key and all the current users' public keys when the membership of the group is changed.

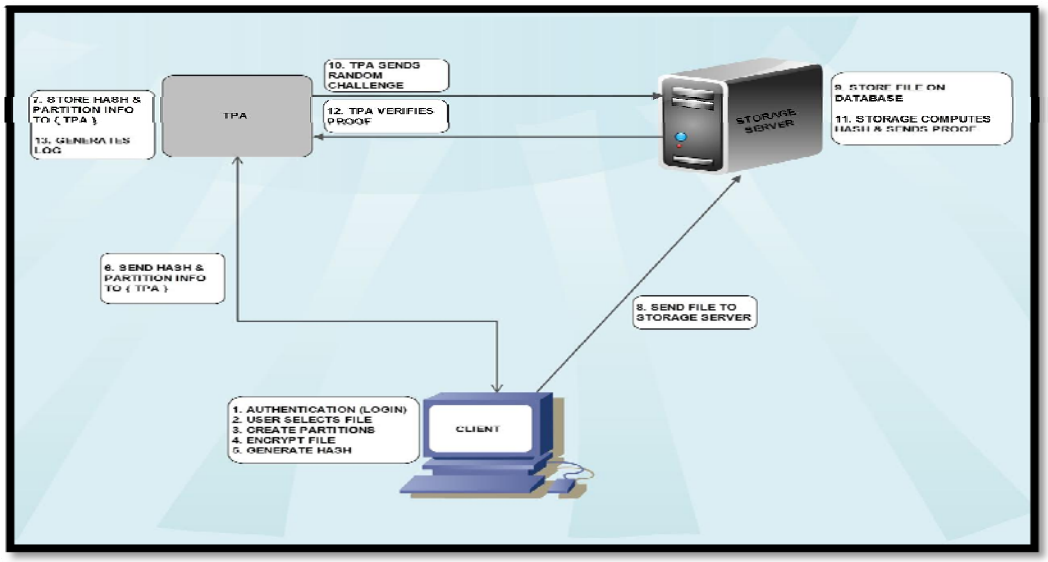


Fig. 1 Secret Sharing of Data

IV. DATA FRESHNESS VERIFICATION

We extend the construction of oruta with data freshness in the authenticated file system that verify the freshness of any data retrieved from the file system while performing typical file system operations. Freshness ensures that the latest version of the data is always retrieved (and thus prevents rollback attacks reverting the file system state to a previous version. Another challenge is efficient management and caching of the authenticating information. Freshness verification should be extremely efficient for existing file system operations and induce minimal latency. To ensure freshness, it is necessary to authenticate not just data blocks, but also their versions. Each block has an associated version counter that is incremented every time the block is modified. This version number is bound to the file-block's MAC: To protect against cloud replay of stale file-blocks (rollback attacks), the counters themselves must be authenticated.

V. FUTURE WORK

In this paper, we propose Oruta, a privacy-retaining public auditing mechanism for shared data in the cloud. We utilize ring signatures to construct homomorphic authenticators, so that a public verifier is able to audit shared data integrity without retrieving the entire data, yet it cannot distinguish who is the signer on each block. To improve the efficiency of verifying multiple auditing tasks, we further extend our mechanism to support batch auditing. There are two interesting problems we will continue to study for our future work. One of them is traceability, which means the ability for the group manager (i.e., the original user) to reveal the identity of the signer based on verification metadata in some special situations. Since Oruta is based on ring signatures, where the identity of the signer is unconditionally protected , the current design of ours does not support traceability. To the best of our knowledge, designing an efficient public auditing mechanism with the capabilities of preserving identity privacy and supporting traceability is still open. Another problem for our future work is how to prove data freshness (prove the cloud possesses the latest version of shared data) while still preserving identity privacy.

VI. CONCLUSION

In this paper, we propose a privacy-retaining public auditing with data freshness verification mechanism for shared data in the cloud. Freshness verification should be extremely efficient for existing file system operations and induce minimal latency.

To ensure freshness, it is necessary to authenticate not just data blocks, but also their *versions*. Each block has an associated version counter that is incremented every time the block is modified. This version number is bound to the file-block's MAC: To protect against cloud replay of stale file-blocks (rollback attacks), the counters themselves must be authenticated.

ACKNOWLEDGEMENT

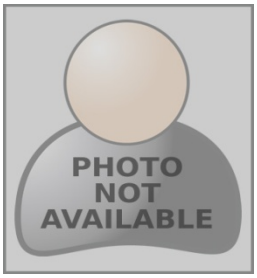
With immense pleasure, I am presenting this seminar report on the " Privacy-Retaining Public Analyzing for Shared Data in the Cloud" as a part of the curriculum of M.E. of Computer Engineering at G.S.Moze College of Engineering. It gives me proud privilege to be complete this survey work under the valuable guidance of Prof. Srinu Dharavath and I would like to extend my thanks to Prof. Ratnaraj Kumar sir (H.O.D of Computer Engineering). I am also be extremely grateful to the Principal for providing all facilities and help for smooth progress of the Journal work. I would also like to thank all the Staff Members of Computer Engineering Dept. Management, friends and my family members who have directly or indirectly guided and who helped me for the preparation of this Journal and gave me an unending support right from the stage the idea was conceived.

References

1. B. Wang, B. Li, and H. Li, "Oruta: Privacy-Preserving Public Auditing for Shared Data in the Cloud," Proc. IEEE Fifth Int'l Conf. Cloud Computing, pp. 295-302, 2012.
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 Computing," Comm. ACM, vol. 53, no. 4, pp. 50-58, Apr. 2010.
3. K. Ren, C. Wang, and Q. Wang, "Security Challenges for the Public Cloud," IEEE Internet Computing, vol. 16, no. 1, pp. 69-73, 2012.
4. D. Song, E. Shi, I. Fischer, and U. Shankar, "Cloud Data Protection for the Masses," Computer, vol. 45, no. 1, pp. 39-45, 2012.
5. C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing," Proc. IEEE INFOCOM, pp. 525-533, 2010.
6. B. Wang, M. Li, S.S. Chow, and H. Li, "Computing Encrypted Cloud Data Efficiently under Multiple Keys," Proc. IEEE Conf. Comm. and Network Security (CNS '13), pp. 90-99, 2013.

AUTHOR(S) PROFILE

Madhubala Rajendra Patil, received the B.Tech degree in Information Technology from V.J.T.I. Mumbai In 2012, now she is pursuing M.E. in Computer Engineering from G. S. Moze College of Engineering, Balewadi in Savitribai Phule Pune University.



Mr Srinu Dharavath is associated with G.S.Moze college of Engineering as Lecturer in Computer Engineering Department.

