International Journal on Applications in Engineering and Technology
Volume 2: Issue 4: April 2016, pp 1-3 www.aetsjournal.com ISSN (Online) : 2455-0523

Dynamic Revocation for shared data verification and data storage in clouds

Amutharasi L, Monica R , Neethu Jayaram, Sowmia R, Geetha.A

Abstract— Cloud users can easily modify and share data as a group. Public auditing in the cloud on the integrity of shared data with these will inevitably reveal confidential information identity privacy to user. In thepaper, novelprivacy-preserving mechanism that supports public auditing on shared data stored in the cloud. In particular, exploit signaturesto compute verification cloud data needed to audit the correctness of shared data. With the mechanism, the identity of the key oneach file in shared data is kept private from public verifiers, who are able to efficiently verify shared data integrity without retrieving the entire file. In addition, the mechanism is able to perform multiple auditing tasks simultaneously instead of verifying them one by one. Experimental results demonstrate the effectiveness and efficiency of the mechanism when auditing shared data integrity is proposed the idea of re-key in group, the cloud to re-key data on behalf of existing users during user revocation, so that existing users do not need to download and re-key data bythemselves.

Keywords— Public auditing, shared data, user revocation, cloud computing.

I. INTRODUCTION

With data storage and sharing services (such as Dropbox and Google Drive) provided by the cloud, peoplecan easily work together as a group by sharing data witheach other. More specifically, once a user creates shareddata in the cloud, every user in the group is able to not onlyaccess and modify shared data, but also share the latest versionof the shared data with the rest of the group. Althoughcloud providers promise a more secure and reliable environmentto the users, the integrity of data in the cloud maystill be compromised, due to the existence of hardware software failures and human errors. To protect the integrity of data in the cloud, a number ofmechanisms have been proposed. In these mechanisms, a signatureis attached to each block in data, and the integrity ofdata relies on the correctness of all the signatures. One ofthe most significant and common features of these mechanismsis to allow a public verifier to efficiently check dataintegrity in the cloud without downloading the entire data,referred to as public auditing (or denoted as Provable Data Possession. This public verifier could be a client whowould like to utilize cloud data for particular purposes(e.g., search, computation, data mining, etc.) or a third-partyauditor (TPA) who is able to provide verification services ondata integrity to users. Most of the previous works focus on auditing theintegrity of personal data. Different from these works, severalrecent works focus on how to preserve identityprivacy from public verifiers when auditing theintegrity of shared data. Unfortunately, none of the abovemechanisms, considers the efficiency of user revocationwhen auditing the correctness of shared data in the cloud.With shared data, once a user modifies a block, shealso needs to compute a new signature for the modifiedblock. Due to the modifications from different users, differentblocks are signed by different users. For securityreasons, when a user leaves the group or misbehaves, thisuser must be revoked from the group. As a result, thisrevoked user should no longer be able to access and modifyshared data, and the signatures generated by thisrevoked user are no longer valid to the group. Therefore,although the content of shared data is not changedduring user revocation, the blocks, which were previouslysigned by the revoked user, still need to be re-signed byan existing user in the group. As a result, the integrity ofthe entire data can still be verified with the public keys ofexisting users only.

Amutharasi.L , Monica.R , Neethu Jayaram , Sowmia.R , Nehru Institute of Technology , Coimbatore-641105, India. (amutharasil@gmail.com), (radhakrishnanmonica@gmail.com) (neethuponnarambil@gmail.com) (sowmiya95ramesh@gmail.com)

Geetha.A, Assistant Professor,Nehru Instiute of Technology,Coimbatore-641105,India.

II. RELATED WORKS

Provable Data Possession (PDP), first proposed by Ateniese etal, allows a public verifier to check the correctness of a client's data stored at an untrusted server. By utilizing RSA-based homomorphic authenticators and sampling strategies, the verifier is able to publicly audit the integrity of data without retrieving the entire data, which is referred to as public verifiability or public auditing. Shacham and Waters designed an improved PDP scheme based on Boneh-Lynn-Shacham (BLS) signatures.To support dynamic operations on data during auditing,Ateniese et al. presented another PDP mechanism based on symmetric keys. However, it is not publicly verifiable and only provides a user with a limited number of verification requests. Wang et al. utilized the Merkle Hash Tree to support fully dynamic operations in a public auditing mechanism. Erway et al. introduced Dynamic Provable Data Possession by using authenticated dictionaries, which are based on rank information. Zhu et al. exploited the fragment structure to reduce the storage of signatures in their public auditing mechanism. In addition, they also usedindex hash tables to provide dynamic operations for users. Wang et al. leveraged homomorphic tokens to ensure the correctness of erasure code-based data distributed onmultiple servers. To minimize the communication overhead in the phase of data

repair, Chen et al. introduced a mechanism for auditing the correctness of data with the multi-server scenario, where these data are encoded with network coding.

More recently, Cao et al. constructed an LT code-based secure cloud storage mechanism. Compared to previous mechanisms , this mechanism can avoid high decodingcomputation costs for data users and save computation resources for online data owners during data repair. Recently,Wang et al. proposed a certificateless public auditing mechanismto reduce security risks in certificate management compared to previous certificate-based solutions.When a third-party auditor is introduced into a public auditing mechanism in the cloud, both the content of data and the identities of signers are private information to users,and should be preserved from the TPA. The public mechanism proposed by Wang et al. is able to preserve users' confidential data from the TPA by using random maskings.In addition, to operate multiple auditing tasks from different users efficiently, they also extended their mechanism to support batch auditing. Our recent work first proposed a mechanism for public auditing shared data in the cloud for a group of users.With ring signature-based on the homomorphicauthenticators, the TPA can verify the integrity ofshared data but is not able to reveal the identity of the signeron each block. The auditing mechanism in is designed to preserve identity privacy for a large number of users. However, it fails to support public auditing.Proofs of Retrievability (POR) is another direction to check the correctness of data stored in a semi-trusted server. Unfortunately, POR and its subsequent work do not support public verification, which fails to satisfy the design objectives in our paper.

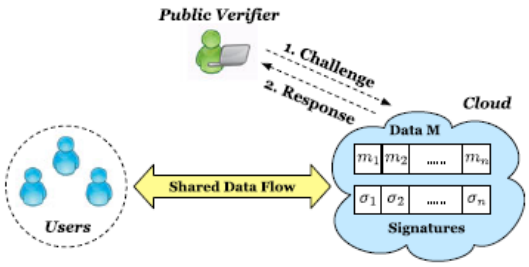
III. EXISTING SYSTEM

In existing mechanisms, a signature is attached to each block in shared data, and the integrity of data relies on the correctness of all the signatures. One of the most significant and common features of these mechanisms is to allow a public verifier to efficiently check shared data integrity in the cloud without downloading the entire data, referred to as public auditing. The public verifier could be a client who would like to utilize cloud data for particular purposes or a third party auditor (TPA) who is able to provide verification services on data integrity to users. With shared data, once a user modifies a block, user also needs to compute a new signature for the modified block. Due to the modifications from different users, different blocks are signed by different users. For security reasons, when a user leaves the group or misbehaves, the user must be revoked from the group. As a result, the revoked user should no longer be able to access and modify shared data, and the signatures generated by the revoked user are no longer valid to the group. Therefore, although the content of shared data is not changed during user revocation, the blocks which were previously signed by the revoked user, still need to be re-signed by an existing user in the group. As a result, the integrity of the entire data can still be verified with the public keys of existing users only.

IV. PROPOSED SYSTEM

Novel public auditing mechanism is proposed for the integrity of shared data with efficient user revocation in the cloud. In the mechanism, by utilizing the idea of proxy re-signatures, once a user in the group is revoked, the cloud is able to resign the blocks, which were signed by the revoked user, with a re-signing key. As a result, the efficiency of user revocation can be significantly improved, and computation and communication resources of existing users can be easily saved. Meanwhile, the cloud, which is not in the same trusted domain with each user, is only able to convert a signature of the revoked user into a signature of an existing user on the same block, but it cannot sign arbitrary blocks on behalf of either the revoked user or an existing user. By designing a new proxy re-signature scheme with nice properties, which traditional proxy re-signatures do not have, our mechanism is always able to check the integrity of shared data without retrieving the entire data from the cloud. Moreover, the proposed mechanism is scalable, which indicates it is not only able to efficiently support a large number of users to share data and but also able to handle multiple auditing tasks simultaneously with batch auditing. In addition, by taking advantages of Shamir Secret Sharing, the mechanism is extended into the multi-proxy model to minimize the chance of the misuse on re-signing keys in the cloud and improve the reliability of the entire mechanism.

V.OVERALL ARHITECTURE



VI. IMPLEMENTATION

System Implementation is the stage in the project where the theoretical design is turned into a working system. The most critical stage is achieving a successful system and in giving confidence on the new system for the user that it will work efficiently and effectively.

The existing system was long time process. The proposed system is developed using C#.net. The existing system caused long time transmission process but the system developed now has a very good user-friendly tool, which has a menu-based interface, graphical interface for the end user.

MD5 algorithm is a widely used cryptographic hash function producing a 128-bit hash value. MD5 is used to verify data intergrity. Message-Digest algorithms are functions

which transform input of arbitrary length into output of constant length.

Public verifier ensures the correctness of data. Public verifier perform intergrity checking without downloading the entire shared data from the cloud. Cloud Service Provider (CSP) has significant storage space.Large data files are uploaded on the remote servers.User can upload files in cloud server after registration.User details are maintained in a database.Revoked user should no longer be able toaccess and modify shared data. Signatures generated by revoked user are nolonger valid to the group. Auditing supports the scalable and efficient public auditing in cloud computing. The public verifier can improve the efficiency of verification by performing audit.

VII. CONCLUSION

In the paper, new public auditing mechanismfor shared data is proposed with efficient user revocation in thecloud. When a user in the group is revoked, the semi-trusted cloud is re-sign blocks that were signed bythe revoked user with proxy re-signatures. Experimentalresults show that the cloud can improve the efficiency ofuser revocation, and existing users in the group can save asignificant amount of computation and communicationresources during user revocation.

REFERENCES

[1] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS'07),pp. 598-610, 2007.

[2] D. Boneh, B. Lynn, and H. Shacham, "Short Signatures from the Weil Pairing," Proc. Seventh Int'l Conf. the Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT'01), pp. 514-532, 2001.

[3] N. Cao, S. Yu, Z. Yang, W. Lou, and Y.T. Hou, "LT Codes-Based Secure and Reliable Cloud Storage Service," Proc. IEEE INFOCOM, pp. 693-701, 2012.

[4] B. Chen, R. Curtmola, G. Ateniese, and R. Burns, "Remote Data Checking for Network Coding-Based Distributed Storage Systems," Proc. ACM Workshop Cloud Computing Security (CCSW'10),pp. 31-42, 2010.

[5] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia, "Dynamic Provable Data Possession," Proc. 16th ACM Conf. Computer and Comm. Security (CCS'09), pp. 213-222, 2009.

[6] H. Shacham and B. Waters, "Compact Proofs of Retrievability," Proc. 14th Int'l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT'08), pp. 90-107, 2008.

[7] B. Wang, B. Li, and H. Li, "Public Auditing for Shared Data with Efficient User Revocation in the Cloud," Proc. IEEE INFOCOM, pp. 2904-2912, 2013.

[8] C. Wang, Q. Wang, K. Ren, and W. Lou, "Ensuring Data Storage Security in Cloud Computing," Proc. 17th ACM/IEEE Int'l Workshop Quality of Service (IWQoS'09), pp. 1-9, 2009.

[9] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling Public Verifiability and Data Dynamic for Storage Security in Cloud Computing," Proc. 14th European Conf. Research in Computer Security(ESORICS'09), pp. 355-370, 2009.

[10] Y. Zhu, H. Wang, Z. Hu, G.-J. Ahn, H. Hu, and S.S. Yau, "Dynamic Audit Services for Integrity Verification of Outsourced Storages in Clouds," Proc. ACM Symp. Applied Computing (SAC'11), pp. 1550-1557, 2011.