ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

ORUTA: Privacy-Preserving Public Auditing For Shared Data in the Cloud

Nahila Zaiba Ruksar
P.G. Scholar (M. Tech),
Department of CSE,
Srinivasa Institute of Technology & Sciences,
Ukkayapalli, Kadapa, Andhra Pradesh.

K. Rajasekhar Reddy
HOD,
Department of CSE,
Srinivasa Institute of Technology & Sciences,
Ukkayapalli, Kadapa, Andhra Pradesh.

Abstract:

With cloud data services, it is commonplace for data to be not only stored in the cloud, but also shared across multiple users. Unfortunately, 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 auditing on the integrity of shared data with these existing mechanisms will inevitably reveal confidential information—identity privacy—to public verifiers. In this paper, we propose a novel privacy-preserving mechanism that supports public auditing on shared data stored in the cloud. In particular, we exploit ring signaturesto compute verification metadata needed to audit the correctness of shared data. With our mechanism, the identity of the signer on each block 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, our mechanism is able to perform multiple auditing tasks simultaneously instead of verifying them one by one. Our experimental results demonstrate the effectiveness and efficiency of our mechanism when auditing shared data integrity.

Index Terms:

Public auditing, privacy-preserving, shared data, cloud computing.

1. INTRODUCTION:

Cloud concept is nothing but the storage service, but it can also share across multiple users. we firstly prioritizes rivacy preserving mechanism because while auditing data from cloud services it's not a secured while that private information is publicly protected by cloud service.

Specifically, the group signature scheme enables users to anonymously use the cloud resources, and the dynamic broadcast encryption technique allows data owners to securely share their data files with others including new joining users which protects the confidentiality from the revoked users in the dynamic broadcast encryption scheme. We propose that while any user is accessing the data from cloud it must be secured by unauthorized person or hacker. Cloud is un-trusted file storage, so we utilize encryption based access control for sharing document in the cloud storage service. User's data is encrypted by using cryptographic technique because unauthorized person can hack the user's private data. In this cryptographic technique we uses different algorithms like signature algorithm, key generation algorithm, ring verify algorithm, etc. these algorithms are used in the cryptographic technique. Users can enjoy high-quality services by migrating local data management systems into cloud servers.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. Besides, many uses of cloud data (e.g., data mining and machinelearning) do not necessarily need users to download the entire cloud data to local devices [2]. It is because cloud providers, such as Amazon, can offer users computation services directly on large-scale data that already existed in the cloud.

II. LITERATURE SURVEY:

A. Privacy-Preserving In The Cloud:

In the Existing system, cloud environment provides large space for storing and managing information for the internet application. The TPA is also important mechanism for authentication is done by this system. The TPA verifies the valid and invalid user by evaluating user identity attributes but if the TPA get hacked by some another then the user not get any notification from cloud due to this users

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

may losses their private information or leakage, so this is big drawback of the existing system. In the previous system, for security purpose OTP (one time password) is not generated while the user's verification is done.

III. RELATED WORK

A. Privacy-Preserving Public Auditing For Shared Data In The Cloud:

In proposed system, we provide Security services including authentication, confident ability and integrity provide in cloud system. In this system we are developed users privacy. The users want to share data from the server then it has insecurity between user and server so TPA application will provide the security to user while he getting the information from the cloud server. The TPA will help us to verify the users correct details and authentication to the server and verifier is able to publicly audit integrity of data without retrieving the entire data.

IV.System Model:

As illustrated in Fig. 2, the system model in this paper involves three parties: the cloud server, a group of users and a public verifier. There are two types of users in a group: the original user and a number of group users. The original ser initially creates shared data in the cloud, and shares it with group users. Both the original user and group users re members of the group.

Every member of the group is allowed to access and modify shared data. Shared data and its verification metadata (i.e., signatures) are both stored in the cloud server. A public verifier, such as a third party auditor providing expert data auditing services or a data user outside the group intending to utilize shared data, is able to publicly verify the integrity of shared data stored in the cloud server.

When a public verifier wishes to check the integrity of shared data, it first sends an auditing challenge to the cloud server. After receiving the auditing challenge, the cloud server responds to the public verifier with an auditing proof of the possession of shared data. Then, this public verifier checks the correctness of the entire data by verifying the correctness of the auditing proof. Essentially, the process of public auditing is a challenge and- response protocol between a public verifier and the cloud server.

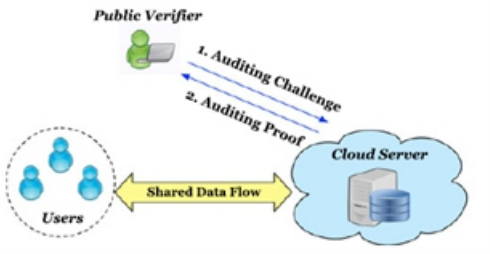

Fig. 1. Our system model includes the cloud server, a group of users and a public verifier.

Threat Model:

Integrity Threats. Two kinds of threats related to the integrity of shared data are possible. First, an adversary may try to corrupt the integrity of shared data. Second, the cloud service provider may inadvertently corrupt (or even remove) data in its storage due to hardware failures and human errors. Making matters worse, the cloud service provider is economically motivated, which means it may be reluctant to inform users about such corruption of data in order to save its reputation and avoid losing profits of its services. Privacy Threats.

The identity of the signer on each block in shared data is private and confidential to the group. During the process of auditing, a public verifier, who is only allowed to verify the correctness of shared data integrity, may try to reveal the identity of the signer on each block in shared data based on verification metadata. Once the public verifier reveals the identity of the signer on each block, it can easily distinguish a high-value target (a particular user in the group or a special block in shared data) from others.

V. SYSTEM ARCHITECTURE:

A. Homomorphic Authrnticable Ring Signature:

How to preserve the users Identity attributes from the TPA because the TPA is un trusted server If the TPA gets hacked by hacker then it may be leakage the users private information so we gave the protection to the server, while it get hack then it will give notification to the user ready to another new user. And again TPA will get ready to another new user.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

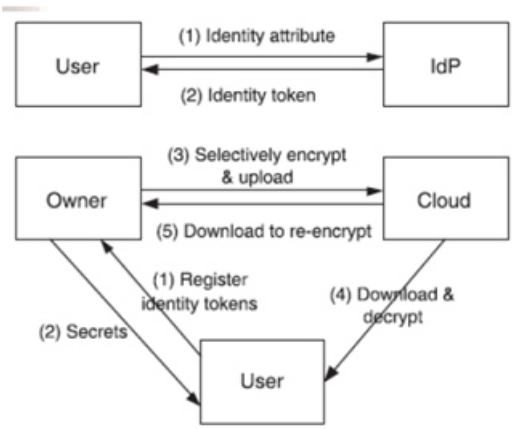

Figure 2 Overall system Architecture

MODERN RING SIGNATURE SCHEME:

Overview: The main motto of ring signatures [12] [13] is to hide the identity of the signer on each block in order to keep private and sensitive information un-disclosed to public verifier. However, the traditional ring signatures does not support block less verifiability and so the verifier needs to download the entire data from the cloud to check the correctness of the shared data which in turn consumes more bandwidth and more time. Therefore, it designs a new homomorphic authenticable ring signature (HARS) scheme, which is extended from classic ring signature scheme. HARS generated ring signatures are not only able to preserve identity privacy but are also able to support block less verifiability.

Construction of HARS:

The HARS contains three algorithms: KeyGen, RingSign and RingVerify. In KeyGen algorithm each user in the group generates his/her public key and private key. In RingSign algorithm 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 identifier is a string; it distinguishes the corresponding block from others. A verifier can check whether a given block is signed by a group member in RingVerify.

PUBLIC AUDITING MECHANISM:

Overview: Using HARS and its properties, a privacy-preserving public auditing mechanism for shared data on cloud is constructed. In this scheme, the public verifier can verify the integrity of shared data without retrieving the entire data.

The identity of the signer on each block in shared data is kept private from the public verifier during the auditing.

Reduce Signature Storage:

Another important issue need to consider in the construction of this scheme is the size of storage used for ring signatures. By the taxonomy of the ring signatures in HARS, a block m is an element of Zp and its ring signature contains d elements of G1, where G1 is a cyclic group with order p. It means a |p|-bit block requires a d * |p| -bit ring signature, which forces users to spend a huge amount of space on storing ring signatures. It will be very frustrating for users, because cloud service providers such as Amazon, will charge users based on the storage space they use.

To reduce the storage of ring signatures on shared data and still allow the public verifier to audit shared data efficiently, we exploit an aggregated approach to expand the size of each block in shared data into k *|p| bits. With the aggregation of a block, the length of a ring signature. is only d/k of the length of a block. Generally, to obtain a smaller size of a ring signature than the size of a block, it choose k > d. As a trade-off, the communication cost of an auditing task will be increasing with an increase of k.

Support Dynamic Operations:

To enable each user in the group to easily modify data in the cloud, there is a need to support dynamic operations on shared data. Dynamic operation such as insert, delete or update operation are performed on a single block. Since the computation of a ring signature includes an identifier of a block, traditional methods which only use the index of a block as its identifier are not suitable for supporting dynamic operations on shared data efficiently.

When a user modifies a single block in shared data by performing an insert or delete operation, the indices of blocks are changed after the block modification and the changes of these indices require users, who are sharing the data, to re-compute the signatures of these blocks, even though the content of these blocks are not modified. This mechanism can allow a user to efficiently perform a dynamic operation on a single block, and avoid the re-computation of indices on other blocks.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Batch Auditing:

Sometimes, a public verifier may need to verify the correctness of multiple auditing tasks in a very short time. Directly verifying these multiple auditing tasks separately would be inefficient. By leveraging the properties of bilinear maps, the concept of batch auditing can be supported, which can verify the correctness of multiple auditing tasks simultaneously and improve the efficiency of public auditing.

Ring Signatures:

The concept of ring signatures was first proposed by Rivest et al. [28] in 2001. With ring signatures, a verifier is convinced that a signature is computed using one of group members' private keys, but the verifier is not able to determine which one. More concretely, given a ring signature and a group of d users, a verifier cannot distinguish the signer's identity with a probability more than 1=d. This property can be used to preserve the identity of the signer from a verifier. The ring signature scheme introduced by Boneh et al. [21] (referred to as BGLS in this paper) is constructed on bilinear maps. We will extend this ring signature scheme to construct our public auditing mechanism.

Construction of Oruta:

Now, we present the details of our public auditing mechanism. It includes five algorithms: KeyGen, SigGen, odify, ProofGen and ProofVerify. In KeyGen, 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. Proof Gen is operated by a public verifier and the cloud server together to interactively generate a proof of possession of shared data. In ProofVerify, 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 pre-defined 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.

We will discuss the case of dynamic groups later Discussion. In the construction of Oruta, we support data privacy by leveraging random masking which is also used in previous work [5] to protect data privacy for personal users. If a user wants to protect the content of private data in the cloud, this user can also encrypt data before outsourcing it into the cloud server with encryption techniques, such as the combination of symmetric key encryption and attribute-based encryption (ABE) With the sampling strategy [9], which is widely used in most of the public auditing mechanisms, a public verifier can detect any corrupted block in shared data with a high probability by only choosing a subset of all blocks(i.e., choosing c-element subset J from set ½1; n_) in each auditing task. Previous work [9] has already proved that,given a total number of blocks n ¼ 1;000;000, if 1 percent of all the blocks are lost or removed, a public verifie can detect these corrupted blocks with a probability greater than 99 percent by choosing only 460 random blocks. Of course, this public verifier can always spend more communication overhead, and verify the integrity of data by choosing all the n blocks in shared data. Even if all the n blocks in shared data are selected (i.e., without using sampling strategy), the communication overhead during public auditing is still much more smaller than retrieving the entire data from the cloud [9]. Besides choosing a larger number of random blocks, another possible approach to improve the detection probability is to perform multiple auditing tasks on the same shared data by using different randoms (i.e., yj is different for block mj in each different task). Specifically, if the current detection probability is Px and a number of t auditing tasks is performed.

VI.RESULTS:

Figure3.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Figure 4.

VII.CONCLUSION AND FUTURE WORK:

In this paper, we propose Oruta, a privacy-preserving public audit ing 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 [21], 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.

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.

[7] R. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public Key Cryptosystems," Comm. ACM, vol. 21, no. 2, pp. 120-126, 1978.

[8] The MD5 Message-Digest Algorithm (RFC1321). https://tools. ietf.org/html/rfc1321, 2014.

[9] 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.

[10] 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.