IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 4, April 2015.
www.ijiset.com
ISSN 2348 – 7968

Reliable Privacy Preserving Public Auditing for Dynamic Groups in Cloud Computing

Devanga Nikita Jakkaiyan[1], M. Murali[2]

[1] Research Scholar, Department of Computer Science, SRM University, Kattankulathur, Tamilnadu, India

[2] Asst. Professor, Department of Computer Science, SRM University, Kattankulathur, Tamilnadu, India

Abstract

The Cloud Utilizers gets scalable, secure and reliable environment from the Cloud Service Providers. Many Cloud Utilizers are and their data storage is subject to mistrust and scrutiny because data in an un-trusted cloud can be easily misplaced, corrupted or vanished due to integrity threats. Hence, in order to maintain the truthfulness of cloud, we need one Third person who will audit all the data which are stored in cloud. Such a person is called as Third Party Auditor (TPA). TPA provides auditing services to the users who give request to do so. PDP (Provable Data Possession) and WWRL (Wang et al.) are two mechanism which supports public auditing and data privacy but do not support identity privacy and blockless verification. Hence the identity privacy in dynamic group is achieved through the mechanism One Ring to Rule Them All (ORUTA). In this proposed mechanism Third Party Auditor is able to verify the truthfulness of shared data for a dynamic group of users without retrieving the whole data and also the user's identity is also kept private from the TPA.ORUTA also supports batch auditing and dynamic operations during public auditing.

Keywords: public auditing, user revocation, dynamic group, share data, cloud computing

1. Introduction

The Cloud Storage services, which is common place where not only data stored over it but also shared among users who are using simultaneously. Enterprise-class infrastructure is provided by the Cloud service provider to cloud utilizers. Some of the cloud offerings are Dropbox and Google Docs. The integrity of cloud services are subject to mistrust and scrutiny, because of the threats. Threats reveal the important information about the shared data or the user who is working on it. PDP and WWRL mechanisms are used for data privacy and security of data over cloud. But the problem occurred that how to preserve the identity from the TPA, who is auditing the system on request of the user. The identity may reveal sensitive information like which part of data is a higher valuable or which is user in group or block is special. Existing mechanisms do not perform public auditing over shared data on dynamic group while preserving identity. The previously used mechanism provides only public auditing services and data privacy only. The mechanism Provable Data Possession (PDP) [2] provides public auditing service only. PDP mechanism allows a verifier to check the fineness of a client's data stored at an un-trusted server. RSA- based encryption method and sampling strategies are utilized by this PDP. The verifier needs not to retrieve whole data and he can easily audit the truthfulness of shared data, which is referred to as public auditing. PDP does not support identity privacy. The mechanism Wang et al. (WWRL) [5] proposed public auditing mechanism which is able to protect the secret data from Third Party Auditor by utilizing random masking technique. In this mechanism the private content which belongs to a personal user is not published to the third party auditor. It supports only public auditing and data privacy. Identity privacy is not achieved in WWRL.

One of the new privacy preserving mechanisms which give the identity privacy in public auditing is ORUTA. To construct ORUTA, Homomorphic Authenticable Ring Signatures (HARS) are utilized. Using these ring signatures TPA is able to verify the integrity of shared data for a dynamic group of users without retrieving the entire data while the identity of the signer on each block in shared data is kept private from the TPA. ORUTA supports batch auditing, where multiple blocks of data can be simultaneously audited. It also uses random masking operation and index hash tables in order to support the dynamic operations like insert, delete and update over the shared data for dynamic group. When a user in the group leaves or misbehaves, the group needs to revoke this user. Here the proposed system also allows the original user to modify the signature of the user who left the group. Original user can create re signature on the blocks which are signed by the revoked user. [13]

ISSN 2348 – 7968

In the next upcoming part of paper organized as follows: Section 2 describes about the existing system model and threat model. Section 3, describes about the Ring Signature and HARS model. The proposed method is detailed Section 4. Section 5 briefly discusses about related work, and finally conclusion of this paper in Section 6.

2. Problem Statement

2.1 Threat Models

Nowadays, in cloud services the integrity of the data is subject to skepticism and scrutiny, because of the threats. Integrity and Privacy are the two main threats considered over Cloud.

a. Integrity Threat: First, an adversary may try to corrupt the integrity of shared data and prevent users from using data correctly. Second, inadvertently the cloud service provider may either corrupt or even remove data in its storage due to hardware failure and human errors. In some cases to avoid jeopardizing its reputation, the cloud server provider may be reluctant to inform users about such corruption of data [1, 14]

b. Privacy Threat: The identity of the signer on each block in shared data is private and confidential to the group. During the process of auditing, Third Party Auditor (TPA), who is responsible for auditing the integrity of shared data, may try to reveal the identity of the signer on each block in shared data based on verification information. Once the TPA reveals the identity of the signer on each block, it can easily define that specific block contains a high-value target or a particular user in the group plays important role on over it [1, 14].

2.2 System Model

The system model mainly includes three parties. They are the cloud server, the Third Party Auditor (TPA) and users. Here the user can be of type: the data owner and group users. The Fig. 1 shows these three parties in the system.

a. User: Data Owner is the user who initially registers with the cloud server. Data owner will stores the information in the cloud storage. Data owner will accept the all group user's request and share the cloud data to the own group users. Data owner also generates the ring signatures using each user's private key and group's public key. Group Users sends the request to the data owner to use the cloud storages. After response from the data owner the group user can use the cloud data through owner. Based on the access control policies the group user can access and modify the data which are created by the data owner.

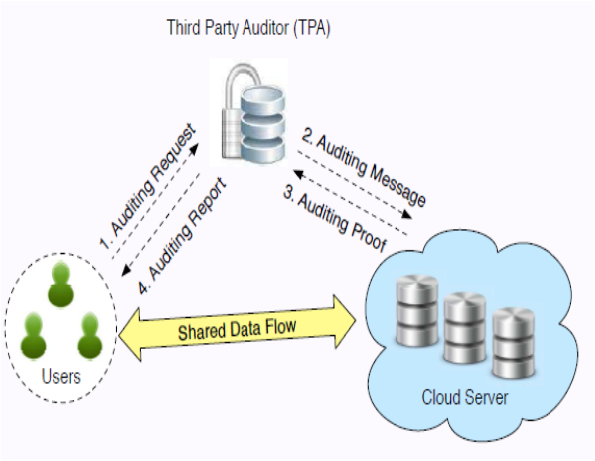

Fig. 1 System model includes the User, Cloud Server and Third Party Auditor.

b. Cloud Server: Data owner uploads the data and the generated ring signature or verification details in Cloud Storage facility. Cloud server contains all the details of the cloud utilizers.

c. Third party Auditor: TPA receives auditing request from the users. Whenever any user wishes to check the integrity of their shared data in cloud, they send the auditing request to the TPA. Public Auditor is any independent third person, having much more computation and communication power than normal user. On behalf of the user TPA performs the auditing process and generates the auditing proof. After verifying the proof he sends the auditing result back to the user.

2.3 Design Objectives

Oruta is designed to achieve the following properties:

1. **Public Auditing:** Without retrieving the entire data, the third party auditor is able to publicly verify the integrity of data for dynamic group of users even if some data chunks are resigned by the cloud.
2. **Reliable User Revocation:** Once user gets revoked from group the block can be efficiently resigned by the existing user. Only the existing users in the group can generate valid signatures on data, and the revoked user can no longer be allowed to compute valid signatures on shared data.
3. **Correctness:** Any corrupted block of data is shared or not is audited by the third party auditor.
4. **Unforgeability:** Only users who exist in the group can generate valid signatures on shared data.
5. **Identity Privacy:** The TPA cannot distinguish the identity of the signer on each block in shared data.

ISSN 2348 – 7968

3. Homomorphic Authenticable Ring Signature

3.1 Ring Signature

The concept of ring signature is first proposed by Rivest et al. Using these 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 member private key[7]. Hence this property can be used to preserve the identity of the signer from a verifier. Author Boneh introduced ring Signature Scheme which referred as BGLS. It is constructed from bilinear maps. These Ring signatures are extended in Oruta to construct public auditing mechanism. It supports batch auditing where multiple auditing tasks from different users efficiently made by leveraging aggregate signature [6].

3.2 HARS

HARS is one of the most suitable mechanism used for public auditing. However, traditional ring signatures [4, 5] cannot be directly used into public auditing mechanisms, because these ring signature schemes do not support blockless verification. Without blockless verification, the TPA has to download the whole data file to verify the correctness of shared data. Hence it consumes excessive bandwidth and takes long verification times. In Classic Ring signature scheme 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. Homomorphic Authenticable Ring Signature allows blockless verification. It contains mainly three algorithms KeyGen, RingSign and RingVerify. First each user in the group generates their own public key and private key in KeyGen Process. In RingSign, a user in the group is able to sign a block with her private key and all the group members' public keys. A verifier is allowed to check whether a given block is signed by a group member in RingVerify.

4. Proposed Method

The proposed method mainly contains six algorithms: KeyGen, SigGen, Modify, ReKeyandSign, ProofGen, and ProofVerify. Users can generate their own key pairs of private and public key. Users are able to create the ring signature on shared content of blocks in SigGen. User can perform the dynamic operations like insert, delete or update operation on a block, and also user can compute the new ring signature on the new modified content. ReKeyandSign is used for generating the new signature on block which are left by the revoked user, and also the revoked user can not be able to generate valid signature on the invalid block of contents. Cloud Server and TPA together generate the valid proof for next process of verification in ProofGen**.** The proofs are validated by the TPA and he sends the auditing report back to the user in ProofVerify.

Scheme Details: Let G1, G2 and GT be multiplicative cyclic groups of order p, g1 and g2 be generators of groups G1, G2, respectively. Let e : G1 × G2 → GT be a bilinear map, and φ : G2 → G1 be a computable isomorphism with φ(g2) = g1. There are three hash functions H1: $\{0,1\}^* \rightarrow G_1, H_2 : \{0,1\}^* \rightarrow Z_q$ and h : $G_1 \rightarrow Z_p$. The global parameters are (e, φ, p, q, $G_1, G_2, G_T, g_1, g_2, H_1, H_2, h$). The total number of users in the group is d. Let U denote the group that includes all the d users.

Shared data M is divided into n blocks, and each block m_j is further divided into k elements inZ_p. Therefore, shared data M can be described as an n×k matrix:

$$M = \begin{pmatrix} m_1 \\ \vdots \\ m_n \end{pmatrix} = \begin{pmatrix} m_{1,1} & \cdots & m_{1,k} \\ \vdots & \ddots & \vdots \\ m_{n,1} & \cdots & m_{n,k} \end{pmatrix} \in Z_p^{nXk} \quad (1)$$

KeyGen: For user U_i in the group U, she randomly picks $X_i \in Z_p$ and computes $W_i = g_2^{Xi}$. The user U_i's

Public key is $pk_i = W_i$ and her private key is $sk_i = X_i$. The original user also randomly generates a public Aggregate key **pak** = $(\eta_1 \ldots \eta_k)$, where $\eta_1$are random elements of G_1.

SigGen: Given all the d group members' public keys $(pk_1 \ldots pk_d) = (w_1, \ldots, w_d)$, a block $m_j = (m_{j,1}, \ldots, m_{j,k})$, its identifier id_j , a private key $sk_s$for some s, user u_s computes the ring signature of this block as follows:

1) She first aggregates block m_j with the public aggregate key **pak**, and computes

$$\beta_j = H_1(id_j) \prod_{i=1}^{k} n_l^{m_{j,1}} \in G_1 \quad (2)$$

2) After computing β_j , user u_s randomly chooses $a_{j,i} \in Z_p$ and sets $\sigma_{j,i} = g_1^{a_{j,i}}$, for all i ≠ s. Then she calculates

$$\sigma_{j,s} = \left(\frac{\beta_{j,i}}{\varphi\left(\prod_{i \neq s} wi^{a_{j,i}}\right)} \right)^{\frac{1}{x_s}} \in G_1 \quad (3)$$

Modify: A user in the group modifies the [jth] block in shared data by performing one of the following three operations:

Insert: This user inserts a new block m_j' into shared data. She computes the new identifier of the inserted block m_j as $id_j' = \{ v_j', r_j' \}$. The virtual index $v_j' = (v_{j-1} + v_j)$, 2, and $r_j' = H_2(m_j' \| v_j')$. For the rest of blocks, the identifiers of these blocks are not changed. This user outputs the new ring signature σ_j' of the inserted block m_j' with SigGen, and uploads $\{m_j', id_j', \sigma_j'\}$ to the cloud server. The total number of blocks in shared data increases to n+1.

ISSN 2348 – 7968

Delete: This user deletes block m_j, its identifier id_j and ring signature $\sigma_j^{'}$ from the cloud server. The identifier of the other blocks in shared data is remaining the same. The total number of blocks in shared data decreases to n-1.

Update: This user updates the j-th block in shared data with a new block $m_j^{'}$. The virtual index of this block is remain the same, and $r_j^{'}$ is computed as $r_j^{'} = H_2\ (m_j^{'} \| v_j)$. The new identifier of this updated block is $id_j^{'} = \{ v_j, r_j^{'}\}$. The identifier of other blocks in shared data is not changed. This user outputs the new ring signature $\sigma_j^{'}$ of this new block with SigGen, and uploads $\{m_j^{'}, id_j^{'}, \sigma_j^{'}\}$ to the cloud server. The total number of blocks in shared data in still n.

ReKeyandSign: This algorithm ReKeyandSign will be evoked at any point of auditing process. HAPS [13] Homomorphic Authenticable Proxy Re-Signature scheme is applied when any user is revoked from the group. When any new member joins the group then he can use the signature which is computed by the data owner in the place of revoked user's signature.

The Cloud Server generates a re-signing key $rk_{i\to j}$ as following method. 1) The cloud generates a random r ∈ Z_p and sends it to the user u_i. 2) User u_i sends r/ x_i to user u_j, where$sk_i = x_i$. 3) User u_j sends rx_j/ x_i to the cloud, where $sk_j = x_j$.4) The cloud recovers$rk_{i\to j} = x_j/x_i \in Z_p$.

When user u_i is revoked from the group, the cloud is able to convert signatures of user u_i into signatures of user u_j on the same block of content. The resigning key user here is $rk_{i\to j}$, public key pk_i, signature σ_k, block $m_k$and the identifier of block id_k, the cloud first checks that e(σ_k, g) = e(H(id_k)wm_k, pk_i). If the verification result will occur as 0, the cloud outputs ⊥, otherwise, it outputs

$$\sigma' k = \sigma_k^{rk_{i\to j}} = (H(id_k)w^{m_k})^{\frac{x_i x_j}{x_i}} = (H(id_k)w^{m_k})^{x_j}$$

After re- signing, the data owner removes u_i's id from the signature.

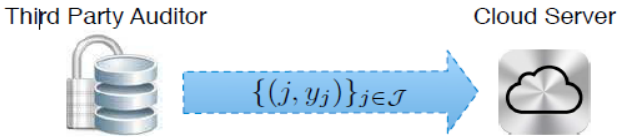

Fig. 2 The TPA sends an auditing message to the cloud server.

ProofGen: As per the Fig. 2, User first sends the auditing request to the TPA. TPA then audits the whole system and generates the valid proof from the ring signatures which are computed in shared block of data. The process will be as follows.

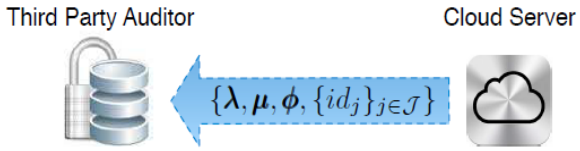

Fig. 3The cloud server sends an auditing proof to the TPA.

1) The TPA randomly picks a c-element subset J of set [1,n] to locate the c selected blocks that will be checked in this auditing process, where n is total number of blocks in shared data.

2) For j ∈ J, the TPA generates a random value $y_j \in Z_q$

Then, the TPA sends an auditing message { (j, y_j) } j ∈ J to the cloud server. After receiving an auditing message {(j, y_j)} j ∈ J, the cloud server generates a proof of possession of selected blocks with the public aggregate key pak.

ProofVerify: With an auditing proof { λ, μ,∅,{ id_j}j ∈ J } , an auditing message {(j , y_j)} j ∈ J, public aggregate key pak = ($\eta_1 \ldots \eta_k$), and all the group member's public keys ($pk_1 .. pk_d$) = (w_1= w_d), the TPA verifies the correctness of this proof by checking the equation in Fig. 3

The TPA checks the proof and TPA believes that the blocks in shared data are all correct, and sends a positive auditing report to the user. Otherwise, TPA sends a negative one.

5. Related Work

The authors M. Armbrust et al. described about the view of Cloud computing. Some threats were explained which allows some adversary to corrupt the system. Because of such threats auditor can easily identify the person and corrupts the data also [1]. Provable Data Possession mechanism was described by Atheniese et al. In order to check the correctness of client's data this mechanism is utilized. 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 [2]. Authors Juels and Kaliski defined model which is able to check the correctness of data. This mechanism is Proofs Of Retrievability (POR) performed over untrusted server [3]. Authors Shacham and Waters designed two Improved POR (Proofs Of Retrievability) schemes. The first scheme the sentinels are built from BLS signatures. In the second scheme POR is based on pseudo-random functions [4]. Authors Wang et al. (WWRL) proposed public mechanism which is able to preserve users' confidential data from

ISSN 2348 – 7968

Third Party Auditor by using random masking [5]. Author Boneh introduced ring Signature Scheme which referred as BGLS. It is constructed from bilinear maps. These Ring signatures are extended in Oruta to construct public auditing mechanism. It supports batch auditing where multiple auditing tasks from different users efficiently made by leveraging aggregate signature [6]. The concept of ring signature is first proposed by Rivest et al. Using these 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 member private key. Hence this property can be used to preserve the identity of the signer from a verifier [7]. To support dynamic operations on data, Ateniese et al. presented an efficient PDP (Provable Data Possession) mechanism based on symmetric keys. One single user is allowed to verify for limited number of times [8]. Author Erway et al. introduced Dynamic Provable Data Possession [DPDP] by using authenticated dictionaries. Ranks are generated for the data which are present over file or block. In public auditing this rank information used to check the integrity of the data is checked [9]. Authors Y.Zhu et al. exploited the fragment structure for the data stored in cloud. In order to reduce the storage of signatures in public auditing mechanism this fragment structure is used [10]. Authors Wang et al. leveraged homomorphic tokens. In multiple servers the codes-based data will be present. Hence to ensure the correctness of such erasure codes-based data these homomorphic tokens are used. This mechanism is able identify the misbehaved servers [11]. Author Tejashree Paigude defined Secure Model for cloud Data Storage (SMDS) where system was considered with trusted Third Party Auditor (TPA) and without TPA. It shows confidentiality of data in two stages. Two stages are i) Data in Rest and ii) and data in transmit. The data present in cloud acts differently in different situation. Therefore they are handled in different ways [12]. Author Boyang Wang and Baochun Li discusses about the proxy re-signature. These re-signatures are computed over the block of data when any user is revoked from the group. The revoked user also can not be able to generate valid signature on un-wanted block of data [13, 15].

6. Conclusions

In order to check the integrity of shared data in dynamic group of users' new public auditing mechanism is defined. Users can revoke from the group due to any malicious behavior or want to exit from group. Hence each time, the auditing should be done with proper ring signature on the shared block of data which are generated by the user. Efficiency is improved in terms of security and storage due to the user revocation, and communicational and computational resources are also improved. The proposed method can also be extended with techniques such as Traceability and Data Freshness.

References

[1] A. M. Armbrust, A. Fox, R. Griffith, A. D.Joseph, R. H.Katz, A. Konwinski,G.Lee, A. Patterson, A.Rabkin, I. Stoica, and M. Zaharia, "*A View of Cloud Computing*", Communications of the ACM, vol.53, no.4, pp 50-58, Apil 2010.

[2] G. Ateniese, R. Burns, R.Curtmola, J. Herring, L. KIssner, Z. Peterson, and D.Song " *Provable Data Possession at Untrusted Stores*," in Proc.ACM Conference on Computer and Communications Security(CCS), 2007, pp 598-610

[3] A.Juels and B.S. Kaliski, "*PORs: Proofs of Retrievability for Large Files*," in Proc. ACM Conference on Computer and Communications Security (CSS), 2007, pp 584-597.

[4] H.Shacham and B.Waters, " *Compact Proofs of Retrievability*, "in Proc. International Conference on the Theory and Application of Cryptograpy and Information Security(ASIACYPT).Springer-Verlag,2008,pp 90-107.

[5] C.Wang, Q. Wang, K. Ren, and W.Lou, "*Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing*," in Proc, IEEE International Conference on Computer Communications(INFOCOM),2010, pp 525-565.

[6] D. Boneh, c. Gentry, B.Lynn, and H. Shacham," *Aggregate and Verifiably Encrypted Signatures from Bilinear Maps*," in Proc. International Conference on the Theory and Applications of Cryptographic Techniques(EUROCRYPT).Springer-Verlag,2003, pp 416-432.

[7] R. L. Rivest, A. Shamir, and Y. Tauman, "*How to Leak a Secret* ,"in Proc. International Conference on the Theory and Application of Cryptography and Information Security(ASIACRYPT).Springer-Verlag,2001.

[8] G.Athniese, R.D. Pietro, L.V.Mancini, and G. Tsudik, "*Scalable and Efficient Provable Data Possession*," in Proc, International Conference on Security and Privacy in Communication Networks (SecureComm),2008, pp 552-565.

[9] G.Athniese, R.D. Pietro, L.V.Mancini, and G. Tsudik, "*Scalable and Efficient Provable Data Possession*," in Proc, International Conference on Security and Privacy in Communication Networks (SecureComm),2008, pp 552-565.

[10] Y.Zhu, H.Wang, Z. Hu, G.-J.Ahn, H. Hu, and S.S.Yau, "*Dynamic Audit Services for Inetgrity Verification of Outsourced Storage in Clouds," In Proc.ACM Symposium on Applied Computing(SAC), 2011, pp 1550-1557.*

[11] C.Wang, Q.Wang, K.Ren, and W.Lou, "*Ensuring Data Storage Security in Cloud Computing*," In Proc. IEEE/ACM International Workshop on Quality of Service (IWQoS), 2009, pp1-9.

[12] Tejashree Paigude, Prof. T.A Chavan "A survey on Privacy Preserving Public Auditing for Data *Storage Security*", International Journal of Computer Trends and Technology, vol.4, 2013.

[13] Boyang Wang, Baochun Li and Hui Li, " *Public Auditing for Shared Data with Efficient User Revocation in the Cloud*", IEEE International Conference on Computer Communications(INFOCOM),2013.

[14] Ms. Sonam, M. Kamble, " *Homomorphic Authenticable Ring Signature(HARS) mechnism for Public Auditing on Shared Data in the Cloud*", International Journal of Engineering Research and General Science Volume 2, Issue 6,2014.

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