Public Auditing of Data Stored in Cloud By Preserving Privacy

Shima V M, Lekshmy D Kumar

Abstract— In cloud computing users can store their data into a cloud server which is located remotely so that users can use high quality applications and services by using available computing resources. The overhead of storing and maintaining local data can be avoided. The problem is that the users no control over their outsourced data makes the integrity of data in cloud server a difficult task. The task is very difficult for users with constrained computing resources. The benefit of cloud computing are those users can use the cloud storage as if it is local. For providing integrity to the data that stored in cloud, users can enable public auditability for cloud storage. Users can resort to a third party auditor (TPA) to check the correctness of their outsourced data and no need to worry about their data integrity. For effective auditing TPA should not introduce any vulnerability. That is user require privacy from the TPA. The auditing method uses homomorphic encryption with random masking technique which provides greater privacy. This paper is based on a secure cloud storage system supporting privacy preserving public auditing.

Index Terms—Cloud computing, Auditing, Batch signature, Multicast authentication etc.

I. INTRODUCTION

Cloud Computing is providing different services through internet by using hardware and software. The services provided by the Cloud computing includes different service models such as platform as a service (PaaS), software as a service (SaaS), Infrastructure as a service (Iaas), storage as a service (StaaS), security as a service (SECaaS), Data as a service (DaaS) etc. The platform as a service (PaaS), software as services (SaaS) Infrastructure as a service (Iaas) is most popular among these. The main advantages of Cloud Computing are: can easily store our data in the cloud without worrying about storage capacity. The data can be accessed from any location, any time on demand. Users only need to pay for what he or she is using. The resources such as hardware or software are easily available without depending on the location.

Security Issues: The security is an important problem in cloud computing. The security of cloud computing is associated with a wide set of policies, to protect data technology & controls deployed, application & the associated infrastructure of cloud computing. The security and privacy problems related with cloud computing are availability, correctness of data, authentication, no data leakage, no data loss, easy maintenance and no storage overhead. In cloud computing, there are mainly two components, a cloud user and a cloud server. Cloud user stores their data in the cloud server. The cloud server is managed by the cloud service provider. The cloud user uploads their data on cloud without worrying about storage capacity and maintenance of their data. The different services are provided by the Cloud Service Provider. The most important problem in cloud data storage is to obtain correctness and integrity of data in the Cloud. No modification or data loss is done. The security of cloud computing can be implemented in different ways for integrity, confidentiality and authentication of cloud data. The integrity or correctness of user's cloud data is another security problem that needs to be considered. The integrity of cloud data can be violated due to a different threat. The cloud service provider may hide the data loss to maintain their reputation.

The major threats in cloud computing is repudiation attack, denial of service attack, spoofing identity theft, information disclosure on up/download intra cloud, data tampering threat. In order to achieve security, the data should be verified by a third outsource party which verifies the correctness or integrity of the user's data. Thus the new concept arrives as Third Party Auditor (TPA) who will audit the user data stored on the cloud, which completely based on the user's request. Thus the cloud service provider doesn't have to worry about the correctness and integrity of the data.

Public key-based homomorphic linear authenticator (or HLA for short) [1],[2], [3] is used in the process of auditing which is a best way to provide privacy while auditing the data stored in the public cloud. The TPA performs the auditing process without demanding the local copy of data and this will reduce the communication and computation overhead as compared to other data auditing approaches. By combining the HLA with random masking the protocol guarantees that the TPA could not learn any information about the data content stored in the cloud server (CS) during the auditing process.

II. RELATED WORKS

A. MAC Based Solution

It is mainly used for data authentication. In this method the user upload both the data blocks and calculated MAC of data to Cloud Server and store the user's secret key to Third Party Auditor. The TPA will retrieve data blocks & the secret key is used to check integrity of the data stored on the cloud. In this method of MAC based solution there are mainly two methods to check integrity of data. A first method is upload the data blocks and the MACs of the data to the cloud server and sends the secret key to the TPA. Then the TPA retrieves data blocks with their MACs and checks the integrity of data using secret key.

Manuscript published on 30 August 2015.
* Correspondence Author (s)

Shima V M*, Match. Student, Sree Chitra Thirunal College of Engineering, Pappanamcode, Trivandrum, India.
Lekshmy D Kumar, Assistant Professor, Department of Computer Science, Sree Chitra Thirunal College of Engineering, Pappanamcode, Trivandrum, India.

The TPA requires the knowledge of the data blocks for verification. For avoiding the requirement of the data to TPA verification, one may restrict the verification to just consist of equality checking.

B. HLA based solution

Homomorphic linear authentication (HLA) is for privacy of data in cloud server [2],[3]. HLA techniques are used to audit data file from cloud server without retrieving the user's data file. HLA generate verification metadata from the user's data file that authenticate the correctness of a data block. That is authenticator is calculated from the linear combination of data blocks. The user authenticates each block of file by a set of HLAs. Then the TPA sends random set of challenge to the cloud server. The cloud server sends back its set of authenticator computed from file blocks.

C. Using Extensible authentication protocol

S. Marium proposed use of Extensible authentication protocol (EAP) using three way hand shake with RSA [4]. For hierarchical architecture they proposed the method of identity based signature. In this an authentication protocol for cloud computing (APCC) is provided. APCC is more efficient and less complex as compared to other authentication protocols. In this method, for authentication Challenge–handshake authentication protocol (CHAP) is used. When a client requests for any service on the cloud, the Service provider authenticator (SPA) sends the first request for identity of client. The steps are as follows

1. SPA sends a CHAP request / challenge to the client when client request for any service to cloud service provider.

2. The Client send back CHAP response/ challenges which is calculated by using any hash function.

3. SPA compares the value of challenge with the calculated value of its own. SPA sends CHAP success message to the client if they are matched. Cloud computing provides authentication of the client by implementing this EAP CHAP method. Spoofing identity theft, data tempering threat, DoS attack can be prevented by using this method. The data is being transferred between client and cloud server. Asymmetric key encryption (RSA) algorithm is used for more providing more security.

III. System Overview

The cloud data storage service involving three different parts, as shown in Fig. 1: the cloud user who stores has large amount of data in the cloud; the Cloud Server provides data storage service and has computation resources and storage space, the third-party auditor which has the responsibility to notify the user about the integrity of the data files stored in the cloud server by performing the important auditing task[5][6]. Cloud users depend on the Cloud Server for cloud data storage and data maintenance. They may also dynamically interact with the CS to access and update their stored data for various application purposes. Users no longer have their data locally. So it is very important for the users to ensure that the integrity of their data is properly maintained. Users cannot perform the correctness verification of data because it causes additional online burden and storage overhead.

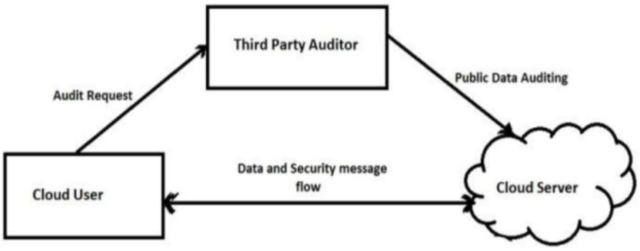


Fig1: Architecture of Public Auditing in Cloud Server

To fully ensure the data correctness and for avoiding additional online burden, it is very important to enable public auditing for cloud data storage. Here users may resort to an independent third party auditor (TPA) to audit the outsourced data whenever needed. The TPA who has the capabilities that can check the integrity of all the data stored in the cloud server which provides an efficient method for the users to ensure their storage integrity in the cloud. Auditing will help users to assess the risk of their cloud data services.

IV. Design and Implementation Of System

The audit performed by TPA would be effective for the cloud service providers for improving their cloud service [5]. For the privacy of data in cloud, the users, who is the owner of the data depends on TPA, the storage security of their data; avoid process of auditing because it introduces additional vulnerabilities of leakage of information causing threat toward their data privacy. By the encryption of data before outsourcing it to cloud is a way to provide privacy which is a concern of data auditing. Encryption is not a complete solution for providing privacy to user's data against third party auditing it is a way to reduce complexity of key management. But there is a chance of leakage of data because of the exposure of key for decryption. Privacy preserving third party auditing protocol is independent to data encryption. For auditing the different TPA is delegated to different user. This problem is addressed by using the technique of public key based homomorphic linear authenticator [7] (or HLA for short), which enables TPA to perform the public auditing without asking the copy of data and thus reduces the communication and computation overhead as compared to the straightforward data auditing approaches.

To achieve privacy preserving public auditing the random masking technique is integrated with the homomorphic authenticator. The server's response is masked with randomness generated by a Pseudo Random Function (PRF) from the linear combination of sampled blocks, since random masking is used and therefore cannot derive the user's data content. There is no need to worry how many linear combinations of the same set of file blocks can be collected. Homomorphic authenticator has the algebraic property. Public key based homomorphic authenticator is used in this scheme [11]. Signature aggregation helps for the multi task auditing [8].

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Consider e:G_1*G_2→G_t be a bilinear map and G_1,G_2,G_t are multiplicative cyclic groups. Assume g be the generator of G_2.

A public auditing scheme consists of different algorithms (KeyGen, SigGen, GenProof, VerifyProof)[4]. A key generation algorithm KeyGen runs to setup the scheme which is done at the user side. SigGen is used by the user to generate verification metadata, which consist of digital signatures. GenProof is run by the cloud server to generate a proof of data storage integrity. VerifyProof is run by the TPA to audit or verify the proof. Running a public auditing system consists of two phases, Setup and Audit[9]:

Setup phase The public and secret parameters are initialized by executing KeyGen, and the data file is preprocessed by using SigGen for generating the verification metadata[1]. The user data file F and the verification metadata stored at the cloud server, and local copy is deleted.

The random signing key pair s_k, s_p are chosen by the cloud user and a random element y is also chosen. The random element belong to the group G1 is denoted as l and then computes m=g^y. The secret parameter s are y, s_k and the public parameter are s_p, m, g, l, e(l, m). Then user computes the authenticator $\sigma_i = (H(W_i).l^{m_i})^y$ where $W_i = n \vee i$, n is the name randomly chosen by the user as the identifier of the particular file. The set of authenticators is stored. The final part of this phase is that file tag is calculated by OR-ing the name n with the signature of name n. Here key used for signature generation is the user's private key. Then the user sends the verification metadata to the server. The verification metadata includes the set of authenticators and the file tag generated by the user. This phase is known as SigGen[1].

Audit phase: The TPA issues a challenge to the cloud server to check the integrity of the data in the cloud server that has to be retained properly at the time of the audit. The cloud server will generate a response message by generating proof using data file and its verification metadata send by the user. This phase is known as GenProof[1]. The TPA then verifies the proof generated by the cloud server. The framework assumes that the TPA is stateless, i.e., TPA does not need to maintain and update state during different audits. This phase is known as VerifyProof[1].

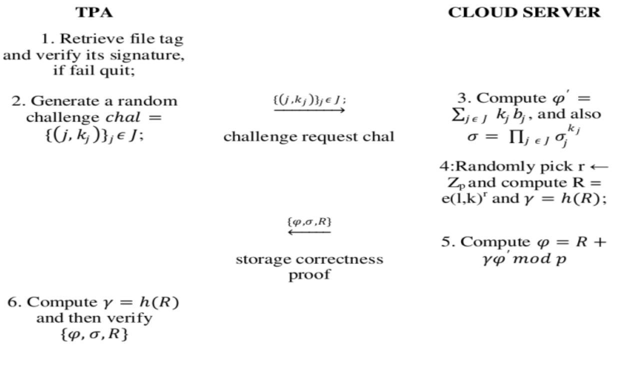

TPA
CLOUD SERVER
1. Retrieve file tag and verify its signature, if fail quit;
2. Generate a random challenge $chal = \{(j, k_j)\}_{j \in J}$;
$\xrightarrow{\{(j,k_j)\}_{j\in J};}$
challenge request chal
3. Compute $\varphi' = \sum_{j \in J} k_j b_j$, and also $\sigma = \prod_{j \in J} \sigma_j^{k_j}$
4:Randomly pick r ← Z_p and compute R = $e(l,k)^r$ and $\gamma = h(R)$;
$\xleftarrow{\{\varphi,\sigma,R\}}$
5. Compute $\varphi = R + \gamma\varphi' \, mod \, p$
storage correctness proof
6. Compute $\gamma = h(R)$ and then verify $\{\varphi, \sigma, R\}$

Table I : The Privacy Preserving Public Auditing Protocol For Cloud Integrity

The TPA retrieves the file tag send by the user and TPA verifies the signature using the public key s_p. If verification succeeds TPA proceeds otherwise quit. TPA issues a challenge message to the cloud server which contains the block position of the file to be checked. Upon receiving the challenge the cloud server calculates the linear combination of sampled blocks and binds it with a random value and sends backs to the TPA. The values are verified by the TPA and thus integrity of the file is maintained. The algorithm is shown below.

Auditing Based on Batch Signature

Multicast authentication can be used in the cloud environment to protect user from malicious attacks[11]. Data integrity, Data origin authentication, Non repudiation are the security services provided by the multicast authentication[12]. The technique here used is an asymmetric key technique called signature. In normal case, the sender generates a signature for each packet with its private key, which is called signing, and each receiver checks the validity of the signature with the sender's public key, which is called verifying. If the verification succeeds, the receiver knows the packet is authentic.

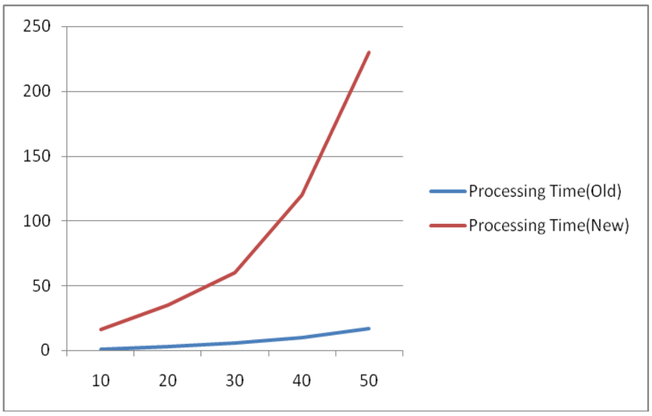
Designing a multicast authentication is a difficult task.

Multicast authentication protocol is also known as MABS (Multicast Authentication based on Batch Signature) contains two schemes. The basic scheme (MABS-B) exploits an efficient asymmetric cryptographic primitive called batch signature, which encourages the authentication of any number of packets simultaneously with one signature verification, to tackle the efficiency and packet loss problems in general environments.

To speed up the auditing of multiple signatures, the batch verification of RSA [8] is used. In order to take advantage of RSA, a sender chooses two large random primes R and S to get M=RS and e,d are estimated which are the two exponents and ed=$1 mod \phi(M)$ where $\phi(M) = (R-1)(S-1)$. The sender advertises its public key (e,M) and keeps private key d in secret. A signature of a message m can be generated as $\sigma = \left(h(m)\right)^d$ mod M where h() is a collision-resistant hash function. The sender sends m, σ to the TPA so that it can verify the cloud server. The file blocks are symbolized as m_i where j=1, 2, 3…n and the corresponding verification metadata is denoted as σ_i . The user side determines the hash of the file block $h_i = h(m_i)$ and convey to the TPA and verification is achieved by the equation.

$$\prod_1^{j=n} \sigma_j = \prod_1^{j=n} h_j \text{modM}$$

V. Results and Analysis



Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

The method that uses homomorphic authenticator takes greater execution time than the batch signature scheme since MABS treats batch signature. The signature size is also greater for the former scheme and later scheme combines the signature into a single signature so that the computation overhead is lesser compared to the other scheme.

VI. CONCLUSION

In this paper, a privacy-preserving public auditing system for data storage integrity in cloud computing is described. The technique used for auditing is homomorphic linear authenticator and random masking. This method guarantees the cloud user that during the efficient auditing process the Third Party Auditor would not learn any details about the content of the file stored on the cloud server. This will eliminates the burden of cloud user from the expensive auditing task and frees the user from the fear about the leakage of outsourced data. Considering TPA concurrently handle multiple audit sessions from different users for their outsourced data files. Further extend the privacy-preserving public auditing protocol into a batch verification scheme Multicast Authentication based on Batch Signature is used where the TPA can perform auditing tasks in a batch manner for better efficiency. Extensive analysis shows that our schemes are provably secure and highly efficient.

REFERENCES

1. W. L. C.Wang, "Privacy-preserving public auditing for storage security in cloud computing," in Proc of IEEE INFOCOM, 2013.
2. .W. H Shacham, "Compact proofs of retrievability," in Proc of Asiacrypt, 2008.
3. G. Ateniese, S. Kamara, and J. Katz, "Proofs of Storage from Homomorphic Identification Protocols," Proc. 15th Int'l Conf. Theory and Application of Cryptology and Information Security: Advances in Cryptology (ASIACRYPT), pp. 319-333, 2009.
4. Q. N. S Marium, "Implementation of eap with rsa for enhancing the security of cloud computing," International Journal of Basic and Applied Science, 2012.
5. 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. 598-609, 2007
6. Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing," IEEE Trans. Parallel and Distributed Systems, vol. 22, no. 5, pp. 847-859, May 2011.
7. Cloud Security Alliance, "Security Guidance for Critical Areas of Focus in Cloud Computing, http://www.cloudsecurityalliance.org, 2009.
8. V. R. D. P K Deshmukh, "Investigation of tpa for cloud data security," International Journal of Scientific and Engineering Research, 2013.
9. C. Wang, Q. Wang, K. Ren, and W. Lou, "Towards Secure and Dependable Storage Services in Cloud Computing," IEEE Trans. Service Computing, vol. 5, no. 2, 220-232, Apr.-June 2012
10. 10. R.C. Merkle, "Protocols for Public Key Cryptosystems," Proc. IEEE Symp. Security and Privacy, 1980.
11. Y. Zhou, X. Zhu, and Y. Fang, "MABS: Multicast Authentication Based on Batch Signature," IEEE Trans.Mobile Computing, vol. 9, pp. 982-993, July 2010
12. 12. K.D. Bowers, A. Juels, and A. Oprea, "HAIL: A High-Availability and Integrity Layer for Cloud Storage," Proc. ACM Conf. Computer and Comm. Security (CCS '09), pp. 187-198, 2009.

Shima V M is currently doing her MTech in Computer Science and Engineering at Sree Chitra Thirunal College of Engineering under University of Kerala, Trivandrum, Kerala, India. Shima received her B Tech Degree in Computer Science and Engineering from Sree Chitra Thirunal College of Engineering under University of Kerala, Kerala, India in 2012. She concentrates mainly on Cloud computing.

Lekshmy D Kumar is working as Assistant professor at the department of computer science and engineering, Sree Chitra Thirunal College of Engineering, Trivandrum, Kerala. She did her B.Tech degree at Sree Chitra Thirunal College of engineering, Trivandrum from University of Kerala. She did her M.Tech degree from NIT, Surathkal. Now she is also doing research in System Analysis and Computer Applications. She published her research works in many international journals.