© August 2015 | IJIRT | Volume 2 Issue 3 | ISSN: 2349-6002

PRIVACY-PRESERVING PUBLIC AUDITING FOR SECURE CLOUD STORAGE

ManoRanjanh[1], N. Sukhendar Reddy[2], Maraty Meena[3]

[1]M.Tech student, CSE Dept, Vishnu sree Institute Of Science and Technology,Telangana, India.

[2]Asst.Professor, CSE Dept Vishnu sree Institute Of Science and Technology, Telangana, India .

[3]Assistant professor ©, CSE, University College of Technology, O.U., (Autonomous), HYD, Telangana.,India.

Abstract- **Cloud computing is an internet based computing which allows sharing of services. Cloud computing allows users to use applications without installation any application and access their personal files and application at any computer with internet or intranet access. Numerous users place their data in the cloud, so exactness of data and security is a prime concern. To ensure the exactness of data, we contemplate the task of allowing a third party auditor (TPA), on behalf of the cloud client, to validate the integrity of the data stored in the cloud., the auditing procedure should carry in no new susceptibilities towards user data privacy, and present no supplementary online burden to user. In this we suggest a secure cloud storage system supporting privacy-preserving public auditing. We promote our result to allow the TPA to perform audits for multiple users concurrently and proficiently. Extensive security and performance analysis show the proposed schemes are provably protected and highly efficient.**

Index Terms— **Homomorphic Linear Authenticator, Third Party Auditor, Public Auditing, Zero knowledge, Data storage, Cloud computing.**

I. INTRODUCTION

Certain cloud services such as Amazon's Simple Storage Service, Box.net, Cloud Safe etc.use user identity, personal data and/or the site of clients.Consequently, these cloud computing services exposed a number of security and privacy apprehensions. The present-day research challenge in cloud services is the secure and privacy-preserving authentication of users. Users, who stock their delicate information like financial information, health records, etc., have a fundamental right of privacy. At hand few cryptographic tools and arrangements like nameless authentication schemes, group signatures, zero knowledge protocols that can together hide user identity and offer authentication. By means of cloud storage, user can distantly store their data and enjoy the on-demand great quality applications and services from a shared pool of configurable computing resources,lacking the load of local data storage and maintenance.Nonetheless, the fact the user on extended have physical controls of the subcontracted data makes the data integrity protection in cloud computing a forbidding task, particularly for the users with unnatural computing resource. Allowing public audit capability for cloud storage is of critical reputation so that user can resort to a third party auditor (TPA) to check the integrity of subcontracted data and be worry-free. To firmly present an operative TPA, the auditing process must carry in no new susceptibilities towards user data privacy, and present no extra online burden to user.Now the secure cloud storage system supporting privacy preserving public auditing is suggested. For instances, CSP might retrieve storage for economic reasons by discarding data that has not been or is seldom accessed, or even hide data loss events so as to sustain a reputation [8]–[10].

II. RELATED WORK

Present several works which deal with general security matters in cloud computing but only few works deal also with user privacy. The authors [1] discover the cost of common cryptographic primitives (AES, MD5,SHA-1, RSA, DSA, and ECDSA) and their sustainability for cloud security purposes. The authors deal with the encryption of cloud storage but do not indication privacy-preserving access to cloud storage. The work [2] employs a pairing based signature scheme BLS to make the privacy-preserving security audit of cloud storage data by the Third Party Auditor (TPA).The resolution uses batch verification to diminish communication overhead from cloud server and computation cost on TPAside. Additional, the paper [3] presents the verification protocols that can lodge dynamic data files. The paper exploresthe problem of providing concurrent public auditabilityand data dynamics for remote data integrity check in CloudComputing in a privacy-preserving way. These solutions [2]and [3] carry privacy-preserving public audit but do notoffer the unspecified access of users to cloud services. Thework [4] establishes requirements for a secure and unspecifiedcommunication system that practicescloud architecture (Tor andFreenet). Nevertheless, the author does not outline any cryptographic solution. An additional non-cryptographic solution confirminguser privacy in cloud consequences is presented in [5]. D. Shrinivas proposed Homomorphic nonlinearauthenticator with arbitrary masking methods to attaincloud security [7].

III. PROPOSED SYSTEM MODEL

Consider a cloud data storage service linking threedissimilar entities, as demonstrated in Fig. the cloud user (U),who has large quantity of data files to be stored in the cloud;the cloud server (CS), which is achieved by the cloud serviceprovider (CSP) to offer data storage service and hassignificant storage space and computation resources, the thirdparty auditor (TPA), who has expertise and skills thatcloud users do not have and is reliable to assess the cloudstorage service dependability on behalf of the user upon request.Users rely on the CS for cloud data storage and maintenance.Assuming that the data integrity threats towards user data canoriginate from both internal and external attacks at CS. These may contain: software bugs, hardware failures, bugs in the network path, economically inspired hackers, malicious or accidental management errors, etc.

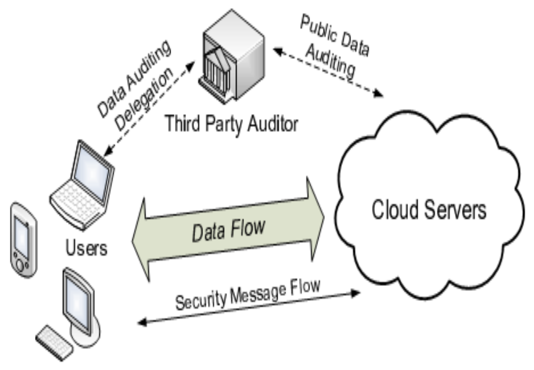


Fig. 1: The architecture of cloud data storage service

In addition, CS can be self interested. Intended for their own benefits, such as to continuereputation, CS might even choose to hide these data corruption events to users. By means of third-party auditing service provides a cost-effective technique for users to gain faith in the cloud. In view of the TPA, who is in the business of auditing, is reliable and autonomous. Nevertheless, it may damage the user if the TPA could learn the subcontracted data later the audit. To approve the CS to reply to the audit delegated to TPA's, the user can issue a certificate on TPA's public key,and entirely audits from the TPA are authenticated contrary to such a certificate.

Design Goals

To allow privacy-preserving public auditing for cloud data storage under the above-mentioned model, our protocol strategy should attain the following security and performance assurances.

1. Public auditability: To permit TPA to verify the accuracy of the cloud data on demand without recovering a copy of the whole data or announcing supplementary online burden to the cloud users.
2. Storage correctness: To confirm that there occurs no dishonest cloud server that can pass the TPA's audit lacking indeed storing users' data intact.
3. Privacy-preserving: To confirm that the TPA cannot originate users' data content from the information collected through the auditing process.
4. Lightweight: To permit TPA to complete auditing with least communication and computation overhead.

THE PROPOSED SCHEMES

This section grants public auditing scheme which delivers a whole outsourcing solution of data and its integrity checking. A public auditing arrangement entails of four algdelivers aKeyGen, SigGen, GenProof, VerifyProof). Key Gen is a key generation algorithm that is run by the user to format the arrangement. Sig Gen is used by the user to produce verification meta data, which may contain of MAC, signatures, or other associated information that will be used for auditing. Gen Proof is run by the cloud server to produce a proof of data storage precision, while VerifyProof is run by the TPA to audit the resistant from the cloud server.

Running a public auditing system entails of two stages,Setup and Audit:

1. Setup: The user prepares the public and secret parameters of the system by performing KeyGen, and preprocesses the data file F by means of

SigGen to produce the verificationmetadata. The user at that moment stores the data file F and the verification metadata on the cloud server, and remove its local copy. As per pre-processing, the user may change;the data file F by escalating it or including extra metadata to be stored on the server.

2. Audit: The TPA issues an audit message or challenge to the cloud server to make certain that the cloud server has recollected the data file F correctly at the time of the audit.The cloud server will originate a reply message from a function of the stored data file F and its confirmation metadata by performing GenProof. The TPA then confirms the response via VerifyProof. Our framework assumes the TPA is stateless, which is a needed property accomplished by our recommended solution.

The TPA is stateless, i.e., TPA does not need to preserve and update state among audits, which is a needed property in the public auditing scheme [6].

The Basic Schemes

The principal one is a MAC-based solution which undergoes from adverse systematic disadvantages – bounded usage and stateful verification,which may pose extra online load to users, in a public auditing situation. The subsequent one is a system based on homomorphic linear authenticators(HLA), which covers far current proofs of storage systems.

MAC-based Solution. Present we had two possible means to make use of MAC to authenticate the data. A unimportant way is just uploading the data chunks with their MACsto the server, and directs the equivalent secret key s_k to the TPA. Later, the TPA can randomly retrieve blocks with their MACs and check the accuracy via s_k.

Nevertheless, it suffers from the following severe disadvantages:

1) The number of times a specific data file can be reviewed is restricted by the number of secret keys that must be fixed a priori.
2) The TPA also has to preserve and keep informed state between audits, i.e.,keep track on the exposed MAC keys.
3) It can only support static data, and cannot resourcefully deal with dynamic data at all.

HLA-based Solution. To efficiently support public auditability lacking to recover the data blocks themselves, the HLA method can be used. However letting efficient data auditing and consuming only constant bandwidth, the direct acceptance of these HLA-based techniques is still not appropriate for our determinations.

Privacy-Preserving Public Auditing Scheme

We suggest to exclusively mixing the homomorphic linear authenticator with arbitrary masking technique. In our protocol, the linear grouping of sampled blocks in the server's response is masked with haphazardness generated the server. With random masking, the TPA no longer has all the essential information to shape up a accurate group of linear equations and consequently cannot derive the user's data content,no matter how many linear groupings of the same set of file blocks can be collected. Alternatively, the correctness validation of the block authenticator couples can still be carried out in a new way which will be shown soon,even with the occurrence of the randomness.

Scheme Details: Let us consider G_1, G_2 and G_T be multiplicative cyclic groups of prime order p, and e: $G_1 \times G_2 \rightarrow G_T$ be a bilinear map as presented in beginnings. Let g be a generator ofG_2. H (•) is a secure map-to-point hash function: $\{0, 1\}^* \rightarrow G_1$, which draws strings consistently to G_1. Additional hash function h (•): $G_T \rightarrow Z_p$draws group element of $G_T$uniformly to Z_p.

The recommended scheme is as follows:

Setup Phase: The cloud user runs KeyGen to create the public and secret parameters. Exactly, the user picks a arbitrary signing key pair (spk, ssk), a arbitrary $x \leftarrow Z_p$, a arbitrary element $u \leftarrow G_1$, and calculates $v \leftarrow g^x$. The secret parameter is sk = (x, ssk) and the public parameters are pk= (spk, v, g, u, e(u, v)).

Assumed a data file F = $(m_1 . . . m_n)$, the user runs SigGen to calculate authenticat or for all block m_i: $\sigma_i \leftarrow (H (W_i) \cdot u^{m}{}_{i})^x \in G_1$ for every i. At this time W_i = name||i and name is selected by the user consistently at random from Z_p as the identifier of file F. Represent the set of authenticators by $\Phi = \{\sigma_i\}\ 1 \leq i \leq n$.

The latter part of SigGen is for confirming the integrity of the exclusive file identifier name. One simple way to do this is to calculate t = name||SSigssk(name) as the file tag for F, where SSigssk(name) is the signature on name under the private key ssk. For straightforwardness, we undertake the TPA knows the number of blocks n. The user at that moment sends F along with the verification meta data (Φ, t) to the server and removes them from local storage.

Audit Phase: The TPA first recovers the file tag t. With reverence to the mechanism we designate in the Setup phase, the TPA confirms the signature SSig$_{ssk}$ (name) via spk, and leaves by releasing FALSE if the verification fails. If not, the TPA recovers name.At this time it comes to the "core" part of the auditing process. To create the challenge message for the audit "chal", the TPAchoices a random c-element subset I = {s$_1$. . . s$_c$} of set [1, n].Meant for each element i Є I, the TPA also picks a random value v$_i$ (of bit length that can be shorter than |p|, as elucidated in[6]). The message "chal" postulates the positions of the blocks that are compulsory to be checked. The TPA directs chal ={(i, v$_i$)} $_{i\text{Є}I}$ to the server, if the verification fails. Or else, the TPA recovers name.

Currently it derives to the "core" part of the auditing process. To create the challenge message for the audit "chal", the TPAchoices a random c-element subset I = {s$_1$. . . s$_c$} of set [1, n].For every element i Є I, the TPA also selects a random value v$_i$ (of bit length that can be smaller than |p|, as described in[6]). The message "chal" stipulates the positions of the chunks that are required to be checked. The TPA sends chal ={(i, v$_i$)} $_{i\text{Є}I}$ to the server.

Upon reception challenge chal = {(i, -i)}i∈I, the server runs GenProof to produce a response proof of data storage exactness. Exactly, the server picks a random element r ← Z$_p$, and computes R = e (u, v) ^{r}ЄG$_T$. Let μ′represent the linear combination of sampled blocks stated in chal: μ′ = ∑i∈Ivimi To blind μ′ with r, the server calculates: μ= r+γμ′ mod p, where γ= h(R) Є Zp. In the meantime, the server also computes a combined authenticator α=πi∈IαviЄG1.It then directs {μ, α, R} as the response proof of storage exactness to the TPA. With the reply, the TPAruns VerifyProof to validate it by first calculating γ= h(R)and then inspection the verification equation.

$$R.e(\sigma^{\gamma},g)\overset{?}{=}e((\prod_{i=s_1}^{s_c}H\left(W_i^{v_i\gamma}\right).u^{u},v)\ \dots\dots\dots\dots..(1)$$

The protocol is demonstrated in Fig. 2. The exactness of the above verification equation can be expounded as follows:

$$R.e(\sigma^{\gamma},g)$$

$$=e\left((u.v)^{r}.e\left(\prod_{i=s_1}^{s_c}H(W_i)u^{m_i}).^{x.v_i}).^{\gamma},g)\right).\right)$$

$$=e\left((u^{r},v).e(\prod_{i=s_1}^{s_c}H(W_i).^{v_i}u^{v_i m_i}).^{\gamma}.g).^{x}\right)$$

$$=e\Big((u^{r},v).e(\textstyle\prod_{i=s_1}^{s_c}H(W_i).^{v_i}).^{\gamma}.u^{u'\gamma},v)\Big)$$

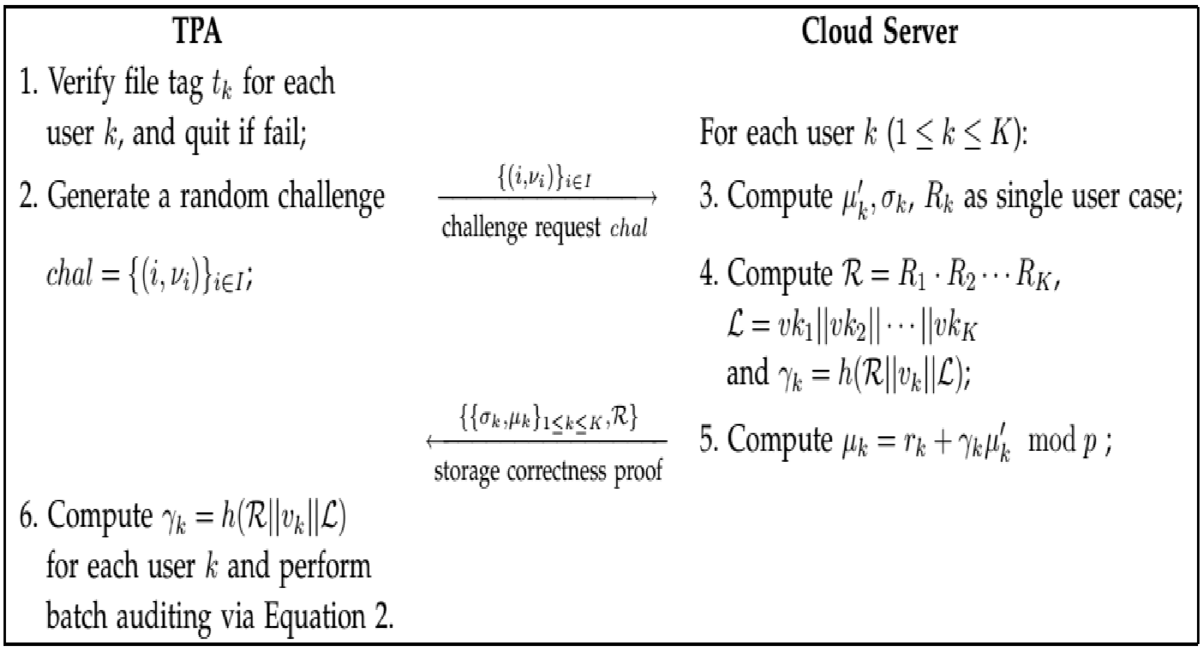
$$=e\left((\prod_{i=s_1}^{s_c}H(W_i).^{v_i}).^{\gamma}u^{u'\gamma+r},v)\right)$$

$$=e\left((\prod_{i=s_1}^{s_c}H(W_i).^{v_i}).^{\gamma}u^{u},v)\right)$$

Properties of the Protocol: It is informal to understand that our protocol attains public auditability. There is no secret keying material or conditions for the TPA to keep or preserve among audits, and the auditing protocol does not pose any potential online load on users. This methodology confirms the privacy of user data content through the auditing process by retaining a random masking r to hideμ, a linear combination of the data blocks.

Reminder that the value R in our protocol, which allows the privacy preserving assurance, will not disturb the validity of the equation, due to the circular association among Rand in γ = h(R) and the verification equation. Storage exactness thus shadows from that of the underlying protocol [6]. As well, the HLA aids attain the constant communication above for server's response during the audit: the size of {μ, α, R} is independent of the number of sampled blocks.

Identification of Invalid Responses. The verification equation (Equation 2) solitary holds as soon as all the responses are valid, and fails with high probability when there is even one particular invalid response in the batch auditing.



TPA

1. Verify file tag t_k for each user k, and quit if fail;
2. Generate a random challenge $chal = \{(i, \nu_i)\}_{i \in I}$;

$\xrightarrow[\text{challenge request } chal]{\{(i,\nu_i)\}_{i\in I}}$

Cloud Server

For each user k $(1 \leq k \leq K)$:

3. Compute μ'_k, σ_k, R_k as single user case;
4. Compute $\mathcal{R} = R_1 \cdot R_2 \cdots R_K$, $\mathcal{L} = vk_1 || vk_2 || \cdots || vk_K$ and $\gamma_k = h(\mathcal{R} || v_k || \mathcal{L})$;
5. Compute $\mu_k = r_k + \gamma_k \mu'_k \mod p$;

$\xleftarrow[\text{storage correctness proof}]{\{\{\sigma_k,\mu_k\}_{1\leq k\leq K},\mathcal{R}\}}$

6. Compute $\gamma_k = h(\mathcal{R} || v_k || \mathcal{L})$ for each user k and perform batch auditing via Equation 2.

Fig. 3: The batch auditing protocol

Support for Data Dynamics

Cutting-edge Cloud computing, subcontracted data might not only be retrieved but also updated frequently by users for various application purposes.Henceforth, supporting data dynamics for privacy preserving public auditing is also of supreme position.

Learning μ′ from σ

However our scheme stops the TPA from straightly deriving μ′ from μ, it does not rule out the likelihood of offline predicting attack from the TPA using valid σ from the response.

IV. CONCLUSON

Here we recommended a privacy-preserving public auditing system for data storage security in Cloud Computing. The homomorphic linear authenticator and random masking promises that the TPA would not study any facts about the data content stored on the cloud server during the efficient auditing process, which not only eradicates the burden of cloud user from the monotonous and perhaps exclusive auditing task, but also eases the users' fear of their out sourced data leakage.

REFERENCES

[1] Y. Chen and R. Sion, "On securing untrusted clouds with cryptography,"in Proceedings of the 9th annual ACM workshop on Privacy in the electronic society. ACM, 2010, pp. 109–114.

[2] C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public cauditing for data storage security in cloud computing," in INFOCOM,2010 Proceedings IEEE, march 2010, pp. 1 –9.

[3] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling public auditability and data dynamics for storage security in cloud computing,"Parallel and Distributed Systems, IEEE Transactions on, vol. 22, no. 5,pp. 847 –859, may 2011.

[4] R. Laurikainen, "Secure and anonymous communication in the cloud,"Aalto University School of Science and Technology, Department of Computer Science and Engineering, Tech. Rep. TKK-CSE-B10, 2010.

[5] M. Mowbray and S. Pearson, "A client-based privacy manager for cloud computing," in Proceedings of the Fourth International ICSTConference on Communication System software and middleware, ser.COMSWARE '09. New York, NY, USA: ACM, 2009, pp. 5:1–5:8.[Online]. Available: http://doi.acm.org/10.1145/1621890.1621897

[6]Shrinivas, "Privacy-Preserving Public Auditing in Cloud Storage security", International Journal of computer science and Information Technologies, vol 2, no. 6, pp.2691-2693, ISSN: 0975-9646, 2011

[7] Jachak K. B., Korde S. K., Ghorpade P. P. and GagareG. J. ,"Homomorphic Authentication with RandomMasking Technique Ensuring Privacy & Security in Cloud Computing", Bioinfo Security Informatics, vol. 2,no. 2, pp. 49-52, ISSN. 2249-9423, 12 April 2012

[8] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner,Z. Peterson, and D. Song, "Provable data

possession at untrusted stores," in Proc. of CCS'07, Alexandria, VA, October2007, pp. 598–609.
[9] M. A. Shah, R. Swaminathan, and M. Baker, "Privacy preserving audit and extraction of digital contents," Cryptology ePrint Archive, Report 2008/186, 2008.
[10] Q. Wang, C. Wang, J. Li, K. Ren, and W. Lou, "Enabling public verifiability and data dynamics for storage security in cloud computing," in Proc. of ESORICS'09, volume 5789 of LNCS.Springer-Verlag, Sep. 2009, pp. 355–370.

BIODATA

AUTHOR 1

ManoRanjan, currently pursing his M.Tech (Computer Science Engg) from VSIT, Bommalaramaram, Nalgonda,Telangana and India.

AUTHOR 2

Narra Sukhendar Reddy,has 6years experience in teaching and currently working as a Sr.Asst.Prof..in CSE Dept at VSIT,Nalgonda,Telangana,India.

AUTHOR 3

Maraty Meena received her B.Tech in ECE in Gokaraju Rangaraju institute of Technology and Sciences, in the year 2006 Miyapur, R.R.Dist, Telangana, and P.G. received in CSE IN JNTU-H, in the year 2011, Hyderabad, Telangana from .Currently working as a Assistant professor(c) in CSE Dept., U.C.T (A)–O.U Hyderabad Telangana.