International Journal of Research (IJR) Vol-1, Issue-7, August 2014 **ISSN 2348-6848**

Auditing Framework for Secure Data Storage in Cloud

A. Mallareddy[1], T.Avinash[2] & K Deepika Rani[3]

1 Research Scholar(JNTUH), Department of Computer Science & Engineering, Professor &HOD(CSE) Sri Indu Institute of Engineering & Technology, Sheriguda(V), Ibrahimpatnam(M), RR Dist – 501510.

2 M.Tech (CSE) , Department of Computer Science & Engineering, Sri Indu Institute of Engineering & Technology, Sheriguda(V), Ibrahimpatnam(M), RR Dist – 501510.

3 Associate Professor, Department of Computer Science & Engineering, Sri Indu Institute of Engineering & Technology, Sheriguda(V), Ibrahimpatnam(M), RR Dist – 501510.

E-mail: ***1 mallareddyadudhodla@gmail.com*** ***2 avinash.talapula9@gmail.com*** ***3. deepikarani.d@gmail.com***

Abstract—

In cloud computing, data owners host their data on cloud servers and users (data consumers) can access the data from cloud servers. Due to the data outsourcing, however, this new paradigm of data hosting service also introduces new security challenges, which requires an independent auditing service to check the data integrity in the cloud. Some existing remote integrity checking methods can only serve for static archive data and, thus, cannot be applied to the auditing service since the data in the cloud can be dynamically updated. Thus, an efficient and secure dynamic auditing protocol is desired to convince data owners that the data are correctly stored in the cloud. In this project, we first design an auditing framework for cloud storage systems and propose an efficient and privacy-preserving auditing protocol. Then, we extend our auditing protocol to support the data dynamic operations, which is efficient and provably secure in the random oracle model.

Keywords: Auditing Framework, Secure Data Storage, Data Storage in Cloud, cloud servers, independent auditing service

I. INTRODUCTION

In cloud computing, data owners host their data on cloud servers and users (data consumers) can access the data from cloud servers. Due to the data outsourcing, however, this new paradigm of data hosting service also introduces new security challenges, which requires an independent auditing service to check the data integrity in the cloud. Some existing remote integrity checking methods can only serve for static archive data and, thus, cannot be applied to the auditing service since the data in the cloud can be dynamically updated. Thus, an efficient and secure dynamic auditing protocol is desired to convince data owners that the data are correctly stored in the cloud. In this project, we first design an auditing framework for cloud storage systems and propose an efficient and privacy-preserving auditing protocol. Then, we extend our auditing protocol to support the data dynamic

operations, which is efficient and provably secure in the random oracle model.

II. CHALLENGES IN CLOUD

A. Data Storage

A cloud storage service provider should base its pricing on how much storage capacity a business has used, how much bandwidth was used to access its data, and the value-added services performed in the cloud such as security. Unfortunately, all the CSPS are not functioning in equal manners‟. Data storage paradigm in “Cloud” brings about many challenging design issues because of which the overall performance of the system get affected. Most of the biggest concerns with cloud data storage are:

Data integrity verification at un-trusted servers:

For example, the storage service provider, which experiences Byzantine failures occasionally, may decide to hide the data errors from the clients for the benefit of their own. What is more serious is that for saving money and storage space the service provider might neglect to keep or deliberately delete rarely accessed data files which belong to an ordinary client. Consider the large size of the outsourced electronic data and the clients constrained resource capability, the core of the problem can be generalized as how can the client find an efficient way to perform periodical integrity verifications without the local copy of data files.

Data accessed by unauthorized users:

The confidentiality feature can be guaranteed by the Owner via encrypting the data before outsourcing to remote servers. For verifying data integrity over cloud servers, researchers have proposed provable data possession technique to validate the intactness of data stored on remote sites.

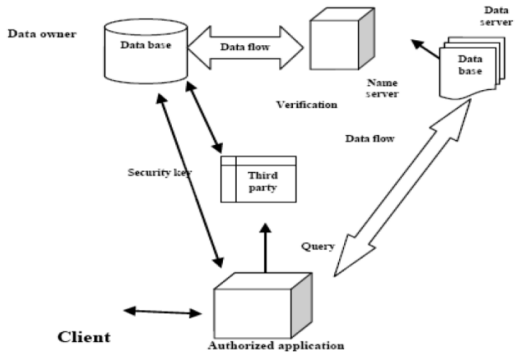


Fig. 1 Storage Cloud

Location Independent Services*:*

The very characteristics of the cloud computing services are the ability to provide services to their clients irrespective of the location of the provider. Services cannot be restricted to a particular location but may be requested from any dynamic location as per the choices of the customer.

Infrastructure and security:

The infrastructure that is used for these services should be secured appropriately to avoid any potential security threats and should cover the life time of component.

Data recovery /Backup:

For data recovery in cloud the user must concern the security as well as the bandwidth issue in consideration..

B. Performance in Cloud

Data storage auditing is a very resource demanding operation in terms of computational resource, memory space, and communication cost. There are three performances criteria in the design of storage auditing protocols:

Low storage overhead: The additional storage used for auditing should be as small as possible on both the Auditor and the cloud server.

Low communication cost: The communication cost required by the auditing protocol should be as low as possible.

Low computational complexity: The computational complexity for storage auditing should be low, especially on the Auditor.

C. Auditing

After In this section, we describe the system model and threat model of data storage auditing protocol in cloud computing. Some models are discussed here:

Data Owner Auditing:

In recent years, with the development of distributed storage systems and online storage systems, the data storage auditing problem becomes even more significant and many protocols have been proposed: e.g., Remote Integrity Checking (RIC) protocols, Proof of Retrievability (POR) protocols and Provable Data Possession (PDP) protocols . However, most of the existing protocols only allowed data owners to check the integrity of their remote stored data. We denote this type of auditing protocols as the Data Owner Auditing.

***Third Party Auditing*:** For the Third Party Auditing, the system model contains three types of entities: data owners, the cloud server and the third party auditor. During the system initialization, data owners compute the metadata of their data and negotiate the cryptographic keys with the third party auditor and the cloud server. Each auditing query is conducted via a challenge-response auditing protocol, which contains three phases: Challenge, Proof and Verification. When the third party auditor wants to check the correctness of data owners‟ data stored on the cloud server, it generates and sends a challenge to the cloud server. The cloud server generates a proof of data storage and sends it back to the third party auditor. Then, the third party auditor runs the verification to check the correctness of the proof from the cloud server and extracts the result on this audit query.

III. BACKGROUND WORK

to allow the auditor to check the data integrity on the remote server, gives the comparisons among some existing remote integrity checking schemes in terms of the performance, the privacy protection, the support of dynamic operations and the batch auditing for multiple owners and multiple clouds. we can find that many of them are not privacy preserving or cannot support the data dynamic operations, so that they cannot be applied to cloud storage systems. In the

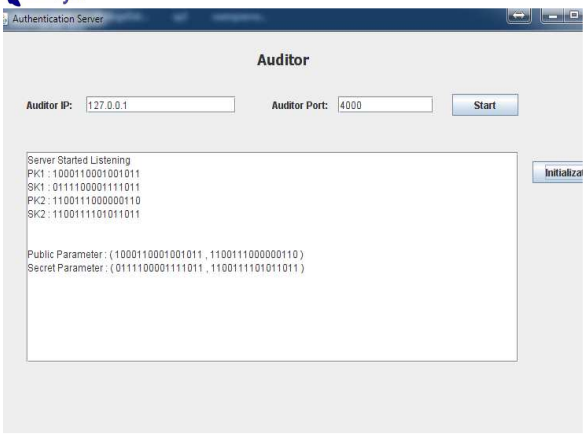
authors proposed a dynamic auditing protocol that can support the dynamic operations of the data on the cloud servers, but this method may leak the data content to the auditor because it requires the server to send the linear combinations of data blocks to the auditor. The authors extended their dynamic auditing scheme to be privacy preserving and support the batch auditing for multiple owners. However, due to the large number of data tags, their auditing protocols may incur a heavy storage overhead on the server. Zhu et al. proposed a cooperative provable data possession scheme that can support the batch auditing for multiple clouds and also extend it to support the dynamic auditing. However, their scheme cannot support the batch auditing for multiple owners. That is because parameters for generating the data tags used by each owner are different, and thus, they cannot combine the data tags from multiple owners to conduct the batch auditing. Another drawback is that their scheme requires an additional trusted organizer to send a commitment to the auditor during the multi cloud batch auditing, because their scheme applies the mask technique to ensure the data privacy. However, such additional organizer is not practical in cloud storage systems. Furthermore, both Wang's schemes and Zhu's schemes incur heavy computation cost of the auditor, which makes the auditor a performance bottleneck.

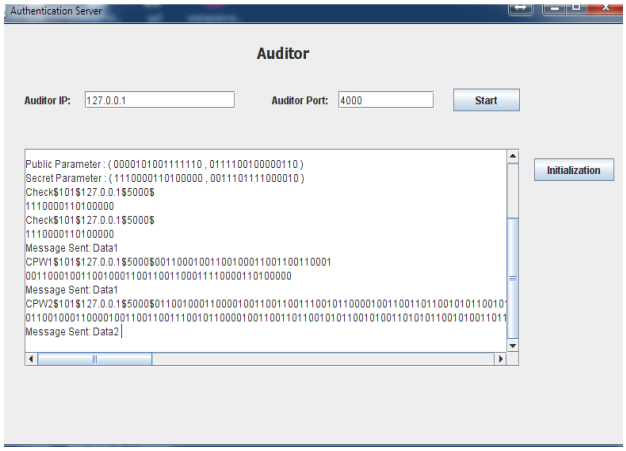
IV. PROPOSED SYSTEM

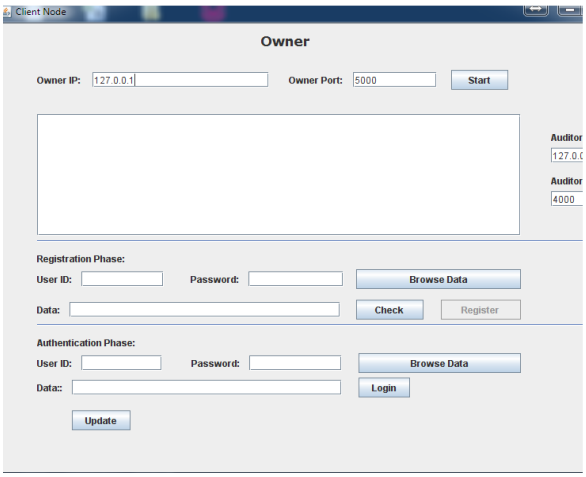
Auditing Framework:

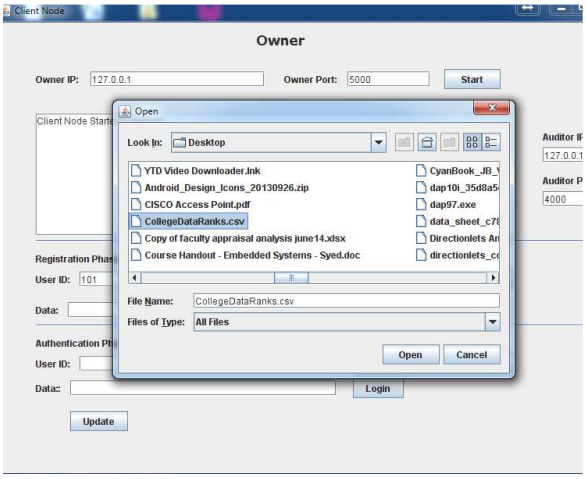
Propose an efficient and secure dynamic auditing protocol, which can meet the requirements. To solve the data privacy problem, our method is to generate an encrypted proof with the challenge stamp by using the Bilinearity property of the bilinear pairing, such that the auditor cannot decrypt it but can verify the correctness of the proof. Without using the mask technique, our method does not require any trusted organizer during the batch auditing for multiple clouds. On the other hand, in our method, we let the server compute the proof as an intermediate value of the verification, such that the auditor can directly use this intermediate value to verify the correctness of the proof. Therefore, our method can greatly reduce the computing loads of the auditor by moving it to the cloud server. Our original contributions can be summarized as follows: We design an auditing framework for cloud storage systems and propose a privacy-preserving and efficient storage auditing protocol. Our auditing protocol ensures the data privacy by using cryptography method and the Bilinearity property of the bilinear pairing, instead of using the mask technique. Our auditing protocol incurs less communication cost between the auditor and the server. It also reduces the computing loads of the auditor by moving it to the server. We extend our auditing protocol to support the data dynamic operations, which is efficient and provably secure in the random oracle model. We further extend our auditing protocol to support batch auditing for not only multiple clouds but also multiple owners. Our multi cloud batch auditing does not require any additional trusted organizer. The multi owner batch auditing can greatly improve the auditing performance, especially in large-scale cloud storage systems.

V. EXPERIMENTAL RESULTS









VI. CONCLUSION

We presented a framework for mining, tracking, and validating evolving multifaceted user profiles on Web sites that have all the challenging aspects of real-life Web usage mining, including evolving user profiles and access patterns, dynamic Web pages, and external data describing ontology of the Web content. A multifaceted user profile summarizes a group of users with similar access activities and consists of their viewed pages, search engine queries, and inquiring and inquired companies. The choice of the period length for analysis depends on the application or can be set, depending on the cross-period validation results. Even though we did not focus on scalability, the latter can be addressed by following an approach similar, where Web click streams are considered as an evolving data stream, or by mapping some new sessions to persistent profiles and updating these profiles, hence eliminating most sessions from further analysis and focusing the mining on truly new sessions.

VII. REFERENCES

[1] G. Ateniese, R.D. Pietro, L.V. Mancini, and G. Tsudik, "Scalable and Efficient Provable Data Possession," IACR Cryptology ePrint Archive, vol. 2008, p. 114, 2008.

[2] C.C. Erway, A. Ku¨ pc¸u¨ , C. Papamanthou, and R. Tamassia, "Dynamic Provable Data Possession," Proc. ACM Conf. Computer and Comm. Security, E. Al-Shaer, S. Jha, and A.D. Keromytis, eds., pp. 213-222, 2009.

[3] Y. Zhu, H. Hu, G. Ahn, and M. Yu, "Cooperative Provable Data Possession for Integrity Verification in Multi-Cloud Storage," IEEE Trans. Parallel and Distributed Systems, vol. 23, no. 12, pp. 2231-2244, Dec. 2012.

[4] 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.

[5] 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, 2010.

[6] K. Yang and X. Jia, "Data Storage Auditing Service in Cloud Computing: Challenges, Methods and Opportunities," World Wide Web, vol. 15, no. 4, pp. 409-428, 2012.

[7] 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, W.C. Chu, W.E. Wong, M.J. Palakal, and C.-C. Hung, eds., pp. 1550-1557, 2011.

[8] C. Wang, K. Ren, W. Lou, and J. Li, "Toward Publicly Auditable Secure Cloud Data Storage Services," IEEE Network, vol. 24, no. 4, pp. 19-24, July/Aug. 2010.

[9] M. Naor and G.N. Rothblum, "The Complexity of Online Memory Checking," J. ACM, vol. 56, no. 1, article 2, 2009.

[10] F. Sebe´, J. Domingo-Ferrer, A. Martinez-Balleste´, Y. Deswarte, and J.-J. Quisquater, "Efficient Remote Data Possession Checking in Critical Information Infrastructures," IEEE Trans. Knowledge Data Eng., vol. 20, no. 8, pp. 1034-1038, Aug. 2008.

[11] 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 Distributed Systems, vol. 22, no. 5, pp. 847-859, May 2011.

[12] 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, J. Pieprzyk, ed., pp. 90-107, 2008.