Privacy Preserving User Auditable Pseudonym System Using Attribute Based Encryption

Renuka Mehta[1] , Aishwarya Khachane[2], Bhagyashree Patil [3], Dhanashree Sonawane [4]
[1, 2, 3, 4] Dept of Electrical and Computer Engineering
[1, 2, 3, 4] SSBT College of Engineering and Technology, Bhambhori, Jalgaon

Abstract- *Cloud computing is the long dreamed vision of computing as a utility, where users can remotely store their data into the cloud so as to enjoy the on-demand high quality applications and services from a shared pool of configurable computing resources. By data outsourcing, users can be relieved from the burden of local data storage and maintenance. Thus, enabling public auditability for cloud data storage security is of critical importance so that users can resort to an external audit party to check the integrity of outsourced data when needed. To securely introduce an effective third party auditor (TPA), the following fundamental requirement have to be met: TPA should be able to efficiently audit the cloud data storage without demanding the local copy of data, and introduce no additional on-line burden to the cloud user. Specifically, our contribution in this work can be summarized as the following three aspects:*

1) *We motivate the public auditing system of data storage security in Cloud Computing and provide a privacy-preserving auditing protocol, i.e., our scheme supports an external auditor to audit user's outsourced data in the cloud without learning knowledge on the data content.*
2) *To the best of our knowledge, our scheme is the first to support scalable and efficient public auditing in the Cloud Computing. In particular, our scheme achieves batch auditing where multiple delegated auditing tasks from different users can be performed simultaneously by the TPA.*
3) *We prove the security and justify the performance of our proposed schemes through concrete experiments and comparisons with the state-of-the-art.*

Keywords- Attribute based encryption, Pseudonym scheme, AES algorithm, Pseudonym Algorithm

I. INTRODUCTION

To enable privacy-preserving public auditing for cloud data storage under the aforementioned model, our protocol design should achieve the following security and performance guarantee:

1) Public auditability: to allow TPA to verify the correctness of the cloud data on demand without retrieving a copy of the whole data or introducing additional on-line burden to the cloud users.
2) Storage correctness: to ensure that there exists no cheating cloud server that can pass the audit from TPA without indeed storing users' data intact.
3) Privacy-preserving: to ensure that there exists no way for TPA to derive users' data content from the information collected during the auditing process.
4) Batch auditing: to enable TPA with secure and efficient auditing capability to cope with multiple auditing delegations from possibly large number of different users simultaneously.
5) Lightweight: to allow TPA to perform auditing with minimum communication and computation overhead.

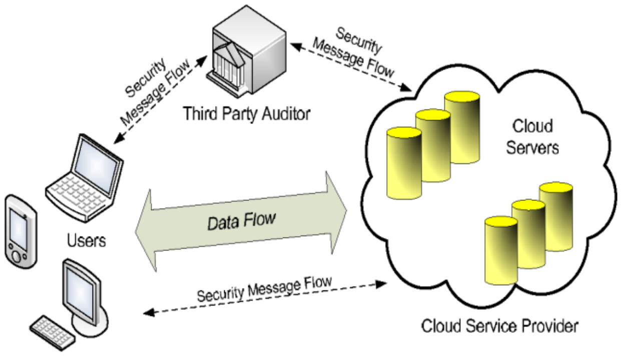

Fig. 1: The architecture of cloud data storage service

Related work

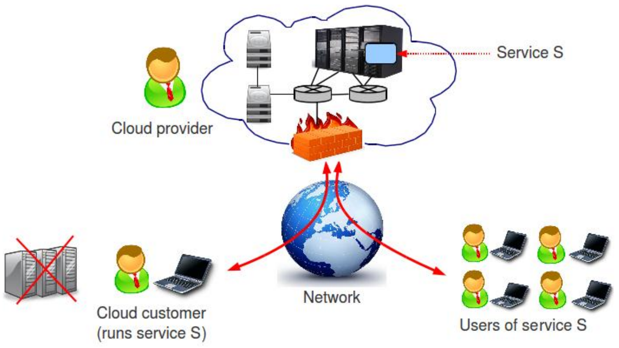

Fig 1: Overall Architecture

II. LITERATURE SURVEY

XingliangYuany,Xinyu Wang Cong Wangy, Chenyun Yu, and SaranaNutanongproposed the similarity search on high dimensional data has been intensively studied for data processing and analytics. Despite its broad applicability, data security and privacy concerns along the trend of data outsourcing have not been fully addressed.

Dr. Jan Camenisch and Dr. Anja Lehmann proposed thethat the pseudonym generate for each user. It takes users identity and provide pseudonym. Pseudonyms are usually taken or adopted to hide an individual ones real identity. A user can encrypt a message under Pseudonym with attributes and time period.

JiayeShao,Yanqin Zhu and QijinJi proposed Attribute based encryption(ABE)which is an efficient technique that exploits attributes and access policies to achieve fine grained access control in cloud computing. Nevertheless, existing multi-authority ABE schemes either can't preserve access policies' privacy or sustain expensive computational cost of encryption and decryption phases. To tackle the above challenges, JiayeShao, YanqinZhu and QijinJi propose an online/offine and outsourced multi authority ABE scheme with policy protection. The main idea is to alleviate the online computation overhead for owners by splitting the encryption algorithm to the online encryption and offine encryption.

III. PROBLEM DEFINITION

To securely introduce an effective third party auditor (TPA), the following two fundamental requirements have to be met:

1) TPA should be able to efficiently audit the cloud data storage without demanding the local copy of data, and introduce no additional on-line burden to the cloud user;
2) The third party auditing process should bring in no new vulnerabilities towards user data privacy.

IV. PROPOSED SYSTEM

In this paper, we utilize the public key based homomorphic authenticator and uniquely integrate it with random mask technique to achieve a privacy-preserving public auditing system for cloud data storage security while keeping all above requirements in mind. To support efficient handling of multiple auditing tasks, we further explore the technique of bilinear aggregate signature to extend our main result into a multi-user setting, where TPA can perform multiple auditing tasks simultaneously. Extensive security and performance analysis shows the proposed schemes are provably secure and highly efficient. We also show how to extent our main scheme to support batch auditing for TPA upon delegations from multi-users.

V. ABE ALGORITHM

The construction of the ABE scheme has the following assumptions:

(a) the user’s identity is used as his/her public key. The message sender uses receiver’s identity to encrypt the message
(b) once receiving the message, the receiver derives the corresponding private key from a trusted third party – the PKG – and decrypts the message
(c) the PKG is in charge of the private key generation. A user must derive the private from the PKG in order to decrypt the ciphertext.

VI. ATTACK MODEL

1. the same plaintext encrypted by the same cryptographic key will generate different ciphertexts
2. The public encryption key is unidentifiable based on known ciphertexts.

VII. PSEUDONYMS GENERATION

1. each AU can self-generate his/her pseudonym and corresponding private key
2. the blind certificate generator (i.e. the organiser) is responsible for the certificate generation and both the pseudonym and the corresponding private key are blind to the organiser.
3. the AUs cannot generate new and valid certificates based on the existing valid certificates
4. the AUs can validate a pseudonym and its corresponding certificate by using the publicly known params.

VIII. KEY GENERATION

1. Derive an arbitrary value i and a permutation key based on master permutation key.
2. Compute the set of randomly-chosen indices:
3. Calculate the token using encoded file and the arbitrary value derived.

IX. KEY VERIFICATION

The procedure of the ith challenge-response for a cross-check over the n servers is described as follows:

i) The user reveals the i as well as the ith key k (i) to each servers.
ii) The server storing vector G aggregates those r rows.
iii) Specified by index k(i) into a linear combination R.
iv) Upon receiving R is from all the servers, the user takes away values in R.
v) Then the user verifies whether the received values remain a valid codeword determined by secret matrix.

X. THIRD PARTY AUDITING PROTOCOL

1. The user blinds each file block data before file distribution k is the secret key for data vector is generated
2. Based on the blinded data vector, the User generates k parity vector via the secret matrix P.
3. The user calculates the ith token for server j as previous scheme
4. The user sends the token secret matrix P, permutation and challenge key Kmaster key, and kchal to TPA for auditing delegation.

XI. ENCRYPTED AUDITING

The general framework followed are six algorithms like Setup, KeyGen, SigGen, Challenge, ProofGen, ProofVerify.

a) Setup : The DO stores the data in cloud by divding it into multiple number of blocks.
b) KeyGen : The DO generates a key using the large prime numbers. Using the large prime number, the public and the secret keys are generated by the DO.
c) SigGen : The DO after generating the keys, will provides its own identity that it is by the true DO, who stored it in cloud.
d) Challenge : Once the keys are signed by the DO, TTPA wants to verify the data. So DO sends a request to TTPA regarding this. The TTPA in turn will send a challenge message to the CSP.
e) ProofGen : Once the CSP receives the challenge message from the TTPA, the CSP generates the proof and responds to TTPA.
f) ProofVerify : On receiving the proof the TTPA verifies the proof and finally it provides result to the data owner.

XII. LIMITATIONS OF EXISTING SYSTEM

1. Accuracy of system is quite less
2. Time consumption increase with increase in dataset size

XIII. ADVANTAGES OF PROPOSED SYSTEM

1. Accuracy is high
2. Time consumption is very less as compared to previous systems
3. Storage is optimized

XIV. DISADVANTAGES OF PROPOSED SYSTEM

1. Does not consider real time data
2. Processing speed depends on the machine configuration

XV. FUTURE SCOPE

1. Can be implemented with other algorithms to check accuracy
2. Hybrid approach can also be implemented to improve accuracy
3. To be implemented using real world data

XVI. CONCLUSION

We propose a privacy-preserving public auditing system for data storage security in Cloud Computing. We utilize the homomorphic linear authenticator and random masking to guarantee that the TPA would not learn any knowledge about the data content stored on the cloud server during the efficient auditing process, which not only eliminates the burden of cloud user from the tedious and possibly expensive auditing task, but also alleviates the users' fear of their outsourced data leakage. Considering TPA may concurrently handle multiple audit sessions from different users for their outsourced data files, we further extend our privacy-preserving public auditing protocol into a multi-user setting, where the TPA can perform multiple auditing tasks in a batch manner for better efficiency. Extensive analysis shows that our schemes are provably secure and highly efficient.

REFERENCES

[1] J. Teevan, E. Adar, R. Jones, and M.A.S. Potts, "Information Re-Retrieval: Repeat Queries in Yahoo's Logs," Proc. 30th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '07), pp. 151-158, 2007.

[2] A. Broder, "A Taxonomy of Web Search," SIGIR Forum, vol. 36, no. 2, pp. 3-10, 2002.

[3] A. Spink, M. Park, B.J. Jansen, and J. Pedersen, "Multitasking during Web Search Sessions," Information Processing and Management, vol. 42, no. 1, pp. 264-275, 2006.

[4] R. Jones and K.L. Klinkner, “Beyond the Session Timeout: Automatic Hierarchical Segmentation of Search Topics in Query Logs,” Proc. 17th ACM Conf. Information and Knowledge Management (CIKM), 2008. P. Boldi, F. Bonchi, C. Castillo, D. Donato, A. Gionis, and S. Vigna, “The Query-Flow Graph: Model and Applications,” Proc. 17th ACM Conf. Information and Knowledge Management (CIKM), 2008.

[5] D. Beeferman and A. Berger, “Agglomerative Clustering of a Search Engine Query Log,” Proc. Sixth ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD), 2000.

[6] R. Baeza-Yates and A. Tiberi, “Extracting Semantic Relations from Query Logs,” Proc. 13th ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD), 2007.

[7] J. Han and M. Kamber, Data Mining: Concepts and Techniques. Morgan Kaufmann, 2000.

[8] W. Barbakh and C. Fyfe, “Online Clustering Algorithms,” Int’l J. Neural Systems, vol. 18, no. 3, pp. 185-194, 2008.

[9] Lecture Notes in Data Mining, M. Berry, and M. Browne, eds. World Scientific Publishing Company, 2006.

[10] V.I. Levenshtein, “Binary Codes Capable of Correcting Deletions, Insertions and Reversals,” Soviet Physics Doklady, vol. 10, pp. 707-710, 1966.

[11] M. Sahami and T.D. Heilman, “A Web-based Kernel Function for Measuring the Similarity of Short Text Snippets,” Proc. the 15th Int’l Conf. World Wide Web (WWW ’06), pp. 377-386, 2006.

[12] J.-R. Wen, J.-Y. Nie, and H.-J. Zhang, “Query Clustering Using User Logs,” ACM Trans. in Information Systems, vol. 20, no. 1, pp. 59-81, 2002.

[13] A. Fuxman, P. Tsaparas, K. Achan, and R. Agrawal, “Using the Wisdom of the Crowds for Keyword Generation,” Proc. the 17th Int’l Conf. World Wide Web (WWW ’08), 2008.