Attribute Based Encryption For Secure Access In Cloud Using Block chaining

Shivapriyan G[#1],Sabareesh Kanna S[#2],Jayashankari J[#3]

[#1],Student, Department of Information Technology, Prince Shri Venketeshwara Padmavathy Engineering College, Ponmar, Chennai

[#2], Student, Department of Information Technology, Prince Shri Venketeshwara Padmavathy Engineering College, Ponmar, Chennai

[#3]Assitant Professor, Department of Information Technology, Prince Shri Venketeshwara Padmavathy Engineering College, Ponmar, Chennai.

***ABSTRACT*: Multi-user system for access control to datasets stored in an untrusted cloud environment. Cloud storage like any other untrusted environment needs the ability to secure share information. Our approach provides an access control over the data stored in the cloud without the provider participation. The main tool of access control mechanism is ciphertext-policy attribute-based encryption scheme with dynamic attributes. Using a blockchain based decentralized ledger, our system provides immutable log of all meaningful security events, such as key generation, access policy assignment, change or revocation, access request. We propose a set of cryptographic protocols ensuring privacy of cryptographic operations requiring secret or private keys. Only ciphertexts of hash codes are transferred through the blockchain ledger. The prototype of our system is implemented using smart contracts and tested on Ethereum blockchain platform.**

INTRODUCTION

Cloud computing encourages users to outsource their data to cloud storage. Data outsourcing means that users lose physical autonomy on their own data, which makes remote data integrity verification become a critical challenge for potential cloud users. To free user from the burden incurred by frequent integrity verifications, Third Party Auditor (TPA) is introduced to perform verifications on behalf of user for data integrity assurance. However, existing public auditing schemes rely on the assumption that TPA is trusted, thus these schemes cannot be directly extended to support the outsourced auditing model, where TPA might be dishonest and any two of the three involved entities (i.e. user, TPA, and cloud service provider) might be in collusion. In this paper, we propose a dynamic outsourced auditing scheme which cannot only protect against any dishonest entity and collision, but also support verifiable dynamic updates to outsourced data. We present a new approach, based on batch-leaves-authenticated Merkle Hash Tree (MHT), to batch-verify multiple leaf nodes and their own indexes all together, which is more appropriate for the dynamic outsourced auditing system than traditional MHT-based dynamism approaches that can only verify many leaf nodes one by one.

Experimental results show that our solution minimizes the costs of initialization for both user and TPA (compared to existing static outsourced auditing scheme), and incurs a lower price of dynamism at user side.

RELATED WORKS:

[1]Revisiting Attribute-Based Encryption With Verifiable Outsourced Decryption. reduce the decryption overhead for a user to recover the plaintext. suggested to outsource the majority of the decryption work without revealing actually data or private keys.provided a requirement of verifiability to the decryption of ABE, but their scheme doubled the size of the underlying ABE ciphertext and the computation costs.

[2]An Algorithmic Approach to Improving Cloud Security: The MIST And Malachi Algorithms.The security algorithms introduced in this paper, the MIST and Malachi are two new ways to protect users' data through account security.

[3]Data Security in Cloud computing and Outsourced Databases.provable data possession (PDP) that allows a client that has stored data at an untrusted server to verify that the server possesses the original data without retrieving it.demonstrate performance superior to other proposals by over 40% on a number of benchmarks.

[4]New Proofs of Retrievability using Locally Decodable Codes.Proofs of retrievability (PoR) are probabilistic protocols which ensure that a client can recover a file he previously stored on a server.Our protocols feature sublinear communication complexity and very low storage overhead.

[5]An Efficient Provable Data Possession Scheme with Data. proposed an efficient Provable Data Possession scheme which uses only hash and symmetric-key cryptographic functions, but it cannot support block insertion.Because of its high efficiency and full dynamics, our scheme is very suitable for applications in which the data needs to be updated after being outsourced.

PROBLEM DEFINITION :

1. Creating User account and assigning user key.

- Input user name and details for the user.
- Generate a random key unique to the user.

2.assigning file keys.

- Users upload files onto the system.
- Generate random key unique to the file.

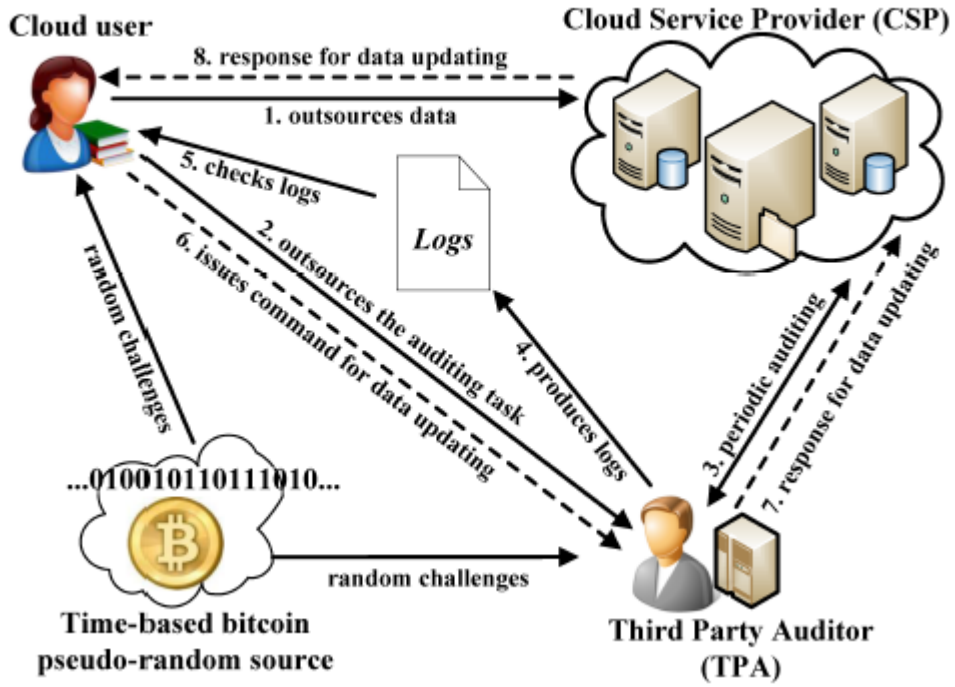
3.display list of all file with attributes such as:

- File name.

- Size.
- Type.

4.share the unique file key in a encrypted form to the user requesting the file .

3.1 SYSTEM ARCHITECTURE :



PROBLEM DESCRIPTION:

1. USER INTERFACE DESIGN: -

To connect with server user must give their username and password. If the user already exits he/she can directly login into the server else user must register their details such as username, password and Email id, into the server. Server will create accounts for every user to maintain upload and download rate. Name will be set as user id. Logging in is usually used to enter a specific page.

2. FILE OWNER UPLOADING:-

Users uploading files or documents into the virtual machines. These constraints serve a dual purpose as they can introduce high-level policies and assist in administration tasks. The user upload the file/data to the cloud.

Given that we rely on network services for our most security-critical data. A source wants to securely send a message to a set of receivers over a cloud network with unit-capacity edges, in the presence of a cloud user.

3. FILE REQUESTING:-

The file is only view format so the file is share and download purpose. File Request is sent to the data owner, the data owner checks the request and id user is authorized, key is provided to the user.

4. THIRD PARTY AUDITOR RESPONSE:-

The malicious cloud might still forge valid authenticators later than the key-exposure time period if it obtains the current secret key of data owner. In this paper, we innovatively propose a paradigm named strong key exposure resilient auditing for secure cloud storage, in which the security of cloud storage auditing not only earlier than but also later than the key exposure can be preserved.

5. FILE RETRIVAL:-

TPA can audit the integrity of the challenged blocks without retrieving these actual blocks from the cloud. But the homomorphic tags can only be computed by user herself to against malicious CSP/TPA. Fortress builds upon the scheme of where the homomorphic tag of data block is constructed by using the corresponding block index.

RESULT:

Thus, we can provide an access control over the data stored in the cloud without the provider participation using ciphertext-policy attribute-based encryption scheme with dynamic attributes.

CONCLUSION:

The outsourced auditing scheme under a stronger security model aims to protect against any dishonest entity and collusion. the new authenticated data structure that is based on Merkle Hash Tree and referred to as BLA-MHT. By supporting the batch-verifications upon multiple leaf nodes, this novel data structure is more efficient than existing MHT-based approaches, and thus is appropriate for the dynamic outsourced auditing system.

FUTURE ENHANCEMENT:

- To provide proxy-based data auditing thus when the file in the cloud have been corrupted the proxy itself enhance protocol to change the corrupted file with original file present in the cloud. Similarly, it provides multiple keyword to check the file stored in cloud.

REFERENCES:

[1] R. Chow, P. Golle, M. Jakobsson, E. Shi, J. Staddon, R. Masuoka, and J. Molina, "Controlling Data in the Cloud: Outsourcing Computation without Outsourcing Control," Proc. 2009 ACM Workshop on Cloud Computing Security (CCSW '09), pp. 85-90, 2009.

[2] Cloud Security Alliance (CSA), "The Notorious Nine Cloud Computing Top Threats in 2013," https://cloudsecurityalliance. org/download/the-notorious-nine-cloud-computing-top -threats-in-2013, Feb. 2013. [

3] G. Ateniese, R.C. Burns, R. Curtmola, J. Herring, L. Kissner, Z.N.J Peterson, and D.X. Song, "Provable Data Possession at Untrusted Stores," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 598-609, 2007.

[4] A. Juels and B.S. Kaliski Jr, "PORs: Proofs of Retrievability for Large Files," Proc. 14th ACM Conf. Computer and Comm. Security (CCS '07), pp. 584-597, 2007.

[5] 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 (ASIACRYPT '08), pp. 90-107, 2008.

[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] C.C. Erway, A. Küpçü, C. Papamanthou, and R. Tamassia, "Dynamic Provable Data Possession," Proc. 16th ACM Conf. Computer and Comm. Security (CCS '09), pp. 213-222, 2009.

[8] D. Cash, A. Küpçü, and D. Wichs, "Dynamic Proofs of Retrievability via Oblivious Ram," Proc. 32nd Int'l Conf. Theory and Applications of Cryptographic Techniques: Advances in Cryptology (EUROCRYPT '13), pp. 279-295, 2013.

[9] C. Wang, S.S.M. Chow, Q. Wang, K. Ren, and W. Lou, "Privacy- Preserving Public Auditing for Secure Cloud Storage," IEEE Trans. Computers, vol. 62, no. 2, pp. 362-375, Feb. 2013.

[10] Y. Zhu, G.J. Ahn, H. Hu, S.S. Yau, H.G. An, and C.J. Hu, "Dynamic Audit Services for Outsourced Storages in Clouds," IEEE Trans. Services Computing, vol. 6, no. 2, pp. 227-238, April-June 2013.