IJREAT International Journal of Research in Engineering & Advanced Technology, Volume 4, Issue 2, April - May, 2016
ISSN: 2320 – 8791 (Impact Factor: 2.317)
www.ijreat.org

Public Auditing For Shared Data In The Cloud Networks With Privacy-Preserving Implementation

Ashutosh Kumar

Tula's Institute,The Engineering & Management College, Uttarakhand

***Abstract* –A cloud administration supplier (CSP) is connected with Third party authority (TPA) for its whole task like information sharing, validation and other reviewing forms. To enhance the productivity of confirmation for numerous examining errands, this research work extends the instrument to support batch scheduling. However, public auditing for such shared data — while preserving identity privacy — remains to be an open challenge. In this, we propose the enhanced privacy-preserving mechanism that allows public auditing on shared data stored in the cloud.**

***Index Terms*— batch scheduling, cloud administration supplier (CSP), public auditing, Third party authority (TPA),**

I. INTRODUCTION

Cloud computing is a model of data taking care of, capacity, and partaking in which quite consolidated physical resources are prepared to remote clients on interest. Rather than getting honest to goodness physical gadgets servers, storage, and frameworks organization stuff clients charge these advantages from a cloud supplier as an outsourced organization that résumés away physical gadgets. By sharing data among tenants, a cloud supplier fulfills economies of scale and equalities workloads, diminishing per-unit resource costs and giving clients the ability to fasten their benefit usage up or down. Cloud computing is versatile and flexible in that it can be gotten to at whatever time from wherever [1].

Utilizing cloud storage; clients can remotely store their data and appreciate the on-interest great applications and administrations from a mutual pool of configurable registering assets, without the weight of nearby data storage and support. In any case, the way that clients no more have physical ownership of the outsourced data makes the data trustworthiness assurance in Cloud computing an impressive undertaking [2].

We aim to accomplish a proficient framework where any approved client can get to the cloud data or database. Furthermore our plan permits composing various times which was not allowed in our before work [3]. We propose secure cloud storage utilizing provable data ownership access control with unknown validation. The records are connected with document access strategies that used to get to the documents set on the cloud. Transferring and downloading of a record to a cloud with standard Encryption/Decryption is more secure. Denial is the imperative plan that ought to expel the records of renounced strategies. So nobody can get to the disavowed record in future. With the framework, the identity of the endorser on each square in shared data is kept private from a Third gathering power (TPA), who is still prepared to unreservedly affirm the genuineness of shared data without recouping the entire archive. The trial results display the reasonability and adequacy of proposed segment when assessing shared data [4].

Objectives of the thesis are as follows:

- To explore such an issue to provide the support of variable-length block verification.
- To propose a scheme to support dynamic scalability on multiple storage servers.
- To propose a scheme that provides all security properties required by zero knowledge interactive proof system, so that it can resist various attacks even if it is deployed as a public audit service in clouds.
- To check authentication scheme in regards of collusion security and protection privacy of the user.

- Moreover to build a provable data possession and robust authentication and access control scheme is.

II. BACKGROUND STUDIES

Background studies are as follows:

A. Wang et al. addressed secure and dependable cloud storage.

Cloud servers inclined to Asymmetric Key Encryption with Privacy Preserving in Clouds Byzantine disappointment, where a capacity server can come up short in subjective ways. The cloud is likewise inclined to data adjustment and server intriguing assaults. In this study clarify the data encryption procedure to store the data in cloud. so the security is high contrasted with alternate plans. Cloud computing offers numerous advantages, however is powerless against dangers. As Cloud computing utilizes build, it is likely that more offenders find better approaches to endeavor framework vulnerabilities [5].

.

B. Adding attributes to role-based access control proposed by D. R. Kuhn, E. J. Coyne, and T. R. Weil

Access Control is any instrument by which a framework concedes or repudiates the privilege to get to data, or perform some activity. Regularly, a client should first Login to a framework, utilizing some Authentication framework. Next, the Access Control instrument controls what operations the client might possibly make by contrasting the User ID with an Access Control database [6].

C. ABS (Attribute Based Signature) is proposed by Maji et al

In this study unknown confirmation is exhibited. For instance, a client might want to store some delicate data yet does not have any desire to be perceived. The client might need to post a remark on an article, however does not need his/her personality to be unveiled. Here clients have a case predicate partner with a message. Claim predicate recognizes the client as an approved one, without uncovering its character. ABS and ABE both are joined to accomplish confirmed access control without revealing the personality of the client to the cloud [7].

D. Zhao et al. provides privacy preserving authenticated access control in cloud

Existing work on access control in cloud are brought together in nature. But, and all different plans utilizes a symmetric key approach and does not bolster verification .And additionally some plans don't bolster confirmation. The current framework has one impediment that is the cloud knows the entrance Verification for every record put away in the cloud [8].

III. PROPOSED WORK

The proposed scheme is in three major steps those are as follows:

A. Proposed Algorithm

Proposed algorithm for block code generation
Input: G is an abelian group;
$g \in G$, m is prime multiplicative order.
Output: A secret $s \in G$ which will be shared by both the sides.
Steps:
Sender generates random $d_A \in \{2,....,m-1\}$ and compute $e_A = g^{d}_{A}$.
Sender sends e_A to receiver.
Receiver generates a random d^B $\{2,....,m-1\}$ and computes $e_B = g^{d}_{B}$.
Receiver sends e_B to receiver.
Sender calculates $s = (e_B)^{d}_{A} = g^{d}_{A}{}^{d}_{B}$
Receiver calculates $s = (e_A)^{d}_{B} = g^{d}_{A}{}^{d}_{B}$

B. Implementation modules:

i. User module:

- **Registration:** In this module each user registers his user details for using files. Only registered user can able to login in cloud server.
- **View Files:**In this module user view a block of uploaded files that is accepted by cloud servers and Verified by verifier in the multi cloud Server.
- **Download:**This module allows the user to download the uploaded from multi cloud server and that file verified by verifier file using his identity key to download the decrypted data.

The client can download the file after completion of the authentication process. As the public key maintained by the key manager, the client request

the key manager for public key. The authenticated client can get the public key. Then the client can decrypt the file with the public key and the private key. The user's credentials were stored in the client itself. During download the file the cloud will authenticate the user whether the user is valid to download the file. But the cloud doesn't have any attributes or the details of the user [9].

ii. Verifier module

- **File Verification module:**The public verifier is able to correctly check the integrity of shared data. The public verifier can audit the integrity of shared data from multi-cloud with whole Data and accept the file [10].

Verification renewal is a tedious process to handle the renewal of the Verification of a file stored on the cloud. Here we implement one additional key called as renew key, which is used to renew the Verification of the file stored on the cloud. The renew key is stored in the client itself [11].

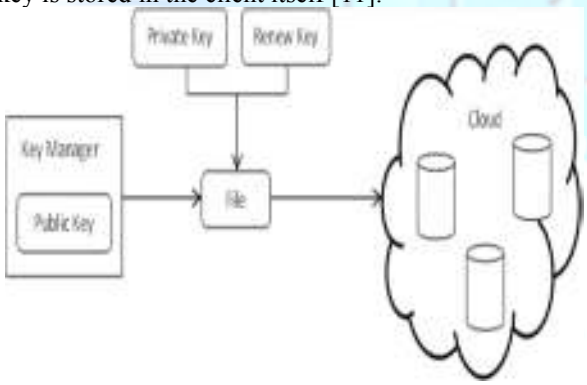

Fig.1: File uploading with data key and renew key [12]

When a file's contract time reaches to expire or Verification has to be revoke on the cloud, there is no need to download all the keys from the cloud. Instead of one renew key is used to revoke the Verification. The client creates a renew key for each file and the keys are encrypted with the control key and fetched with the files, then sent to the cloud [13].

The renewal can be done by the following steps:

1. Download all the encrypted renew keys of each file from the cloud.
2. Send the renew keys to the key manager for decrypt the renew key with the control key.
3. Get the renew keys from the key manager.
4. Generate new renew keys and encrypts with control key.
5. Send the renew keys to the cloud to make the Verification renewal of each file.

- **Accept Files:**In this module public auditor check all files integrity And accept the files to cloud.

C. Owner module

- **Upload files:**The client made request to the key manager for the public key, which will be generated according to the Verification associated with the file. Different policies for files, public key also differs. But for same public key for same Verification will be generated. Then the client generates a private key by combining the username, password and security credentials. Then the file is encrypted with the public key and private key and forwarded to the cloud [14].

iii. Server module

- **Accept files:**Ability to limit and control the access to host systems and applications via communication links. To achieve, access must be identified or authenticated. After achieved the authentication process the users must associate with correct policies with the files.

To recover the file, the client must request the key manager to generate the public key. For that the client must be authenticated. The attribute based encryption standard is used for file access which is authenticated via an attribute associated with the file. With file access control the file downloaded from the cloud will be in the format of read only or write supported. Each user has associated with policies for each file. So the right user will access the right file. For making file access the attribute based encryption scheme is utilized [15].

IV. RESULTS

Results of our proposed technology will be like following below figures:
Run the cloud server file and start the server by passing the server port number. In log you will get Cloud server Amazon S3 started and listening.

Run the TTP file and start the Trusted Third Party by passing the port number. Then connect it to

server by providing cloud server IP address and cloud server port number. In log you will get TTP started and connection established with cloud server.

Fig.2: Run the user file and proceed with registration and login

Create accounts as user and admin (TPA) and login as per respective accounts. Check logs of Cloud server as well as TPA for user connection.Check logs of Cloud server as well as TTP for user connection

Fig.3: Upload file to cloud server and TTP

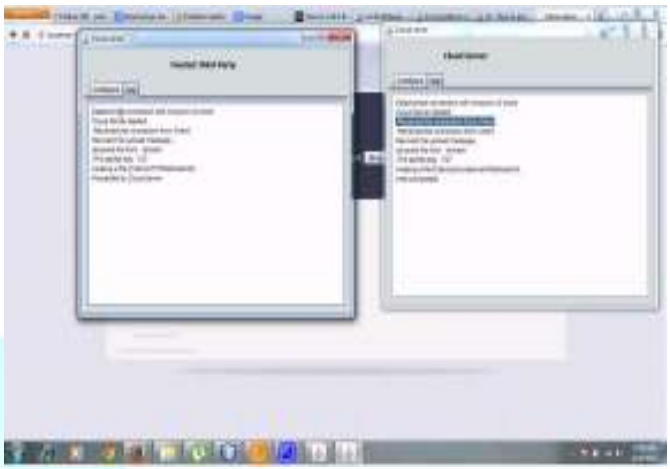

Fig.4: Check logs of Cloud server as well as TTP for file uploaded

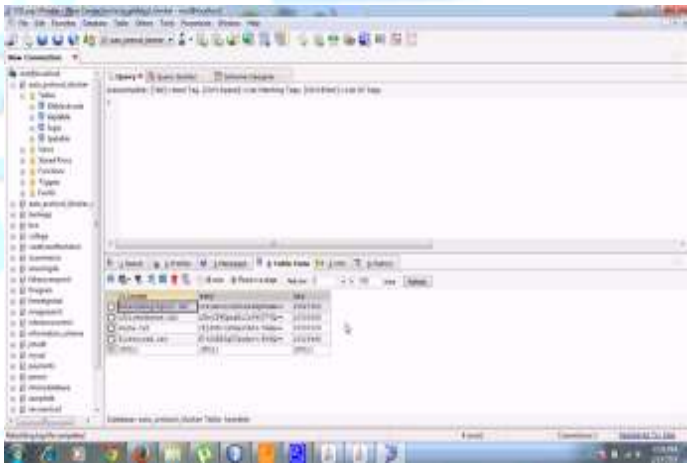

Fig.5: Public the protocol key table in SQLyog Ultimate database

Fig.6: Download file at the user end

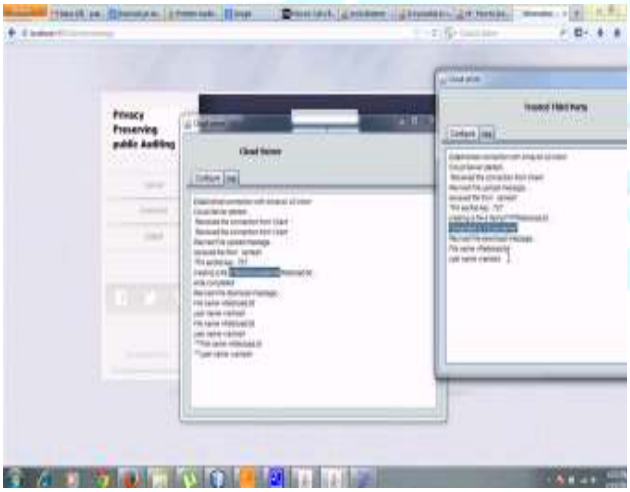

Figure 5.7: Check logs of cloud server and TTP for downloaded file

Figure 5.8: Block audit through TPA to admin using secret code

Figure 5.9: Logs for block verification in TPA

Change the key of protocol blocker to check cheating. Block verification is done by TPA using user name and block code that is generated by server. Block code verifies the document to send the download or upload request to server.

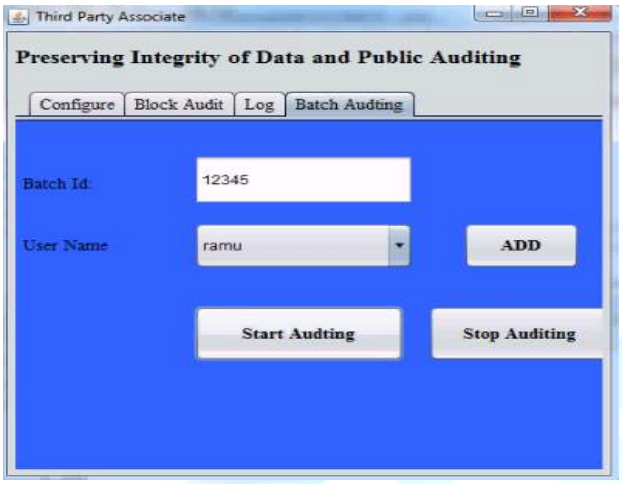

Figure 5.10: Batch scheduling through TPA to files using batch id and user name

Batch scheduling is done by TPA using and batch id that is generated by server. Batch scheduling schedules the multiple documents to send the download or upload request to server

V. CONCLUSIONS

The attribute authority will have the capacity to follow the movement of the client if there should arise an occurrence of negligence with the enlisted Identity of the client. It is likewise feasible for the individual credit power to follow the action of the cloud administration supplier by permitting the ascribe power to have the ID of the CSP enrolled with it. Then again, a trusted party (TTP) can likewise be presented wherein the clients and the CSP register their IDs with the TTP. In the occasion of any deceitful movement the TTP serves as a go between who might track the IDs and distinguish the individual responsible for the fraudulence. In future the file access Verification can be implemented with Multi Authority based Attribute based Encryption.

VI. REFERENCES

[1] SushmitaRuj, Milos Stojmenovic, AmiyaNayak, "Provable data possession Access Control with Anonymous Authentication of data stored in Clouds", in IEEE, 2013.

[2] Boneh, Dan, Giovanni Di Crescenzo, RafailOstrovsky, et Giuseppe Persiano. "Public Key Encryption with Keyword Search", Advances in Cryptology - EUROCRYPT 2004. Springer, 2004.506-522.

[3] C. Gentry, "A fully homomorphic encryption scheme," Ph.D. dissertation, Stanford University, 2009, http: //www. crypto. stanford.

[4] S. Kamara and K. Lauter, "Cryptographic cloud storage," in Financial Cryptography Workshops, ser. Lecture Notes in Computer Science, vol.6054. Springer, pp. 136–149, 2010.

[5] S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in Proceedings of the 29th conference on Data communications, ser. INFOCOM'10. IEEE Press, 2010, pp. 534–542.

[6] R. Kuhn, E. J. Coyne, and T. R. Weil, "Adding attributes to role-based access control," IEEE Computer, vol. 43, no. 6, pp. 79–81, 2010.

[7] K. Maji, M. Prabhakaran, and M. Rosulek, "Attribute-based signatures: Achieving attribute privacy and collusion-resistance," IACR Cryptology ePrint Archive, 2008.

[8] Zhao, T. Nishide, and K. Sakurai, "Realizing fine grained and flexible access control to outsourced data with attribute-based cryptosystems," in ISPEC, ser. Lecture Notes in Computer Science, vol. 6672. Springer, pp. 83–97, 2011.

[9] B. Lewko and B. Waters, "Decentralizing attribute based encryption," in EUROCRYPT, ser. Lecture Notes in Computer Science, vol. 6632. Springer, pp. 568–588, 2011.

[10] Matthew Green, Susan Hohenberger and Brent Waters, "Outsourcing the Decryption of ABE Ciphertexts," in USENIX Security Symposium, 2011.

[11] R. C. Merkle, "Protocols for public key cryptosystems," in Proc.of IEEE Symposium on Security and Privacy, Los Alamitos, CA, USA, 1980.

[12] S. Ruj, A. Nayak, and I. Stojmenovic, "DACC: Cloud access control in clouds," in IEEE TrustCom, 2011.

[13] A.-R. Sadeghi, T. Schneider, and M. Winandy, "Token-based cloud computing," in TRUST, ser. Lecture Notes in Computer

Science, vol. 6101. Springer, pp. 417–429, 2010. edu/craig.

[14] S. D. C. di Vimercati, S. Foresti, S. Jajodia, S. Paraboschi, and P. Samarati, "Over-encryption: Management of access control evolution on outsourced data," in Proceedings of the 33rd International Conference on Very Large Data Bases. ACM, 2007, pp. 123–134.

[15] V. Goyal, O. Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in Proceedings of the 13th ACM conference on Computer and communications security, ser. CCS '06. ACM, 2006, pp. 89–98.