ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 1, January 2017

A Survey on DeyPos for Multiple Users Environment with User Previlege

Shubham Dhole, Arihant Dhumale, Abhisheksingh Dube , Akshay Rathod

B. E Students, Dept. of Comp. Engg., Dr. D. Y. Patil Institute of Engineering & Technology,

Pimpri, Pune(M.S.), India

ABSTRACT: Dynamic Proof of Storage (PoS) could be a helpful scientific discipline primitive that permits a user to see the integrity of outsourced files and to with efficiency update the files in a cloud server. Though scientist have planned several dynamic PoS schemes in unit user environments, the investigation of matter in multi-user environments has not been sufficiently illustrated. A sensible multi-user cloud storage system allows the secure client-side cross-user deduplication technique, that permits a user to skip the uploading method and procure the possession of the files now, once subsequent owners of an equivalent files have uploaded them to the cloud server. No other present system POSs will allow to support this system. In this this paper, we have a capability to introduce the conception of deduplicatable dynamic proof of storage associated propose an economical construction called as DeyPoS, to realize dynamic PoS and secure cross-user duplication, at the same time. By ananlyzing the challenges of structure diversity and personal tag generation, we have a capability to exploit a unique tool called as Homomorphic genuine Tree (HAT). We are enough capable to prove the protection of our construction hence according to theoretical analysis and experimental results, our construction is economical in follow.

KEYWORDS: Deduplication, Proof of ownership, Dynamic proof of storage, Cloud Computing, User Previlege

I. Introduction

Storage outsourcing is evolving into additional and additional enticing to each trade and tutorial because of the advantages of low value, high accessibility, and straightforward sharing. In the recent year the attention gained by the storage outsourcing forms, cloud storage . When user wants to transfer their files to the servers which having many accesses by various devices the firms like Amazon, Google and Microsoft give their own cloud storage to the users. In the current days cloud storage services are wide adopted in current days but there are many security problems and potential threats .Data integrity becomes vital properties when a user outsources its files to cloud storage. Users must convinced that the files keep within the server should not be tampered. There are many Ancient techniques for safeguarding knowledge integrity, like message authentication codes (MACs) and digital signatures need users to transfer all the files from the cloud server for verification that ensures a significant communication value. These techniques are not seem to be appropriate for cloud storage services wherever users could check the integrity oftentimes, like each hour. Hence researchers introduced Proof of Storage (PoS) for checking the integrity while not downloading files from the cloud server. Users might need many dynamic operations, like modification, insertion, and deletion, to update their files also maintaining the potential of PoS. Dynamic PoS is created for such dynamic operations. In addition with PoS, dynamic PoS employ structures looks like the Merkle tree. Hence whenever dynamic operations are dead users regenerate tags for the updated blocks solely instead of creating for all blocks. To increased perceive the subsequent contents. Our tendency to gift additional details concerning PoS and dynamic PoS. The schemes used in our project, each block of a file is hooked up a tag which employed for substantiating the integrity of that block. When a champion confirms the integrity of a file, it selects some block indexes of the file, and sends the files to the cloud server. On consisting the challenge beside their tags the cloud server returns corresponding tags generated.

The index correctness and file integrity is check by the Champion. The direct bonding of tags are done by cryptanalytic tags. There is a path to affect the latter is that the major distinction among PoS and dynamic PoS The indication of correctness and block integrity in the POSs scheme by the block index which is encoded into its tag .

DOI: 10.15680/IJIRCCE.2017. 0501127

Dynamic PoS is not able to cypher the block indexes into tags, since the dynamic operations could modified many indexes of non-updated blocks that incurs reserve computation and communication value. For example, a file consists of one thousand blocks, and a replacement block which is inserted behind the second block of the file. Then, 998 block indexes of the previous first file which are modified that implies the user is able to generate and send 999 tags for this update.For the sake of unravelling this challenges many structure are introduced in dynamic POSs, Result indicate that the tags are hooked up to the structure rather than the block indexes. The reason behind the dynamic PoS remains to be improved in an exceedingly multi-user atmosphere is the necessity of cross-user American state duplication on the client-side. This implies that users will able to skip the uploading method and acquire the ownership of files now as long because the uploaded files present already within the cloud server. This method will reduce back space for storing for the cloud server, and store transmission information measure for users. There are no dynamic PoS that may provide secure cross-user American state duplication.

II. Related work

1] Title: Practical Dynamic Proofs of Retrievability
Authors: Elaine Shi, Emil Stefanov, Charalampos Papamanthou
They propose a dynamic PoR scheme with constant client storage whose bandwidth cost is comparable to a Merkle hash tree, thus being practical. Their construction outputs of the constructions of Stefanov et al. and Cash et al., both in theory and in practice. For n outsourced blocks of β bits each, writing a block requires β +O(β log n) bandwidth and O(β log n) server computation (β is the security parameter). Verification are also efficient, requiring $\beta+O(\beta^2 \log n)$ bandwidth. They also show how to make their scheme publicly verifiable, providing the first dynamic PoR scheme with a property. They finally provide a efficient implementation of their scheme.

2] Title: Provable Data Possession at Untrusted Stores
Authors: Giuseppe Ateniese, Randal Burns, Reza Curtmola, Joseph
They present two provably-secure PDP schemes that are more efficient than previous solutions, even on compared with schemes that achieve weaker guarantees. The problem at the server is low (or even constant), as opposed to linear in the size of the data. Experiments verify the practicality of PDP and reveal that the performance of PDP is bounded by disk I/O and not by cryptographic computation.

3] Title: Attribute-Based Data Sharing Scheme Revisited in Cloud Computing
Authors: Shulan Wang, Kaitai Liang, Joseph K. Liu, *Member, IEEE*, Jianyong Chen, Jianping Yu, and Weixin Xie
They provide an improved two-party key issuing protocol that can guarantee that key authority or cloud service provider can not compromise the whole secret key of a user individually. They introduce the concept of attribute with weight, being provided to improve the expression of attribute, which can not extend the expression from binary to arbitrary state, but also lighten the complexity of access policy. Therefore, both storage cost and encryption complexity for a cipher text are removed. The performance analysis and the security proof show that the proposed scheme can achieve efficient and secure data sharing in cloud computing.

4] Title: Public Auditing For Shared Data With Efficient User Revocation In The Cloud
Authors:Malaneelam Bhaskar , G.Umadevi
They utilize group signatures to construct homomorphic authenticators, so that a third party auditor (TPA) can verify the integrity of shared data for users without receving the entire data.The identity of the signer on each block in shared data is private from the TPA. The amount of information used for verification and the time it takes to verify with it, are not affected by the number of users in the group. In addition, Knox exploits homomorphic MACs to decrease the space used to store such audit information. Their experimental results show that Knox can efficient to audit the correctness of data, shared among a large number of users.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

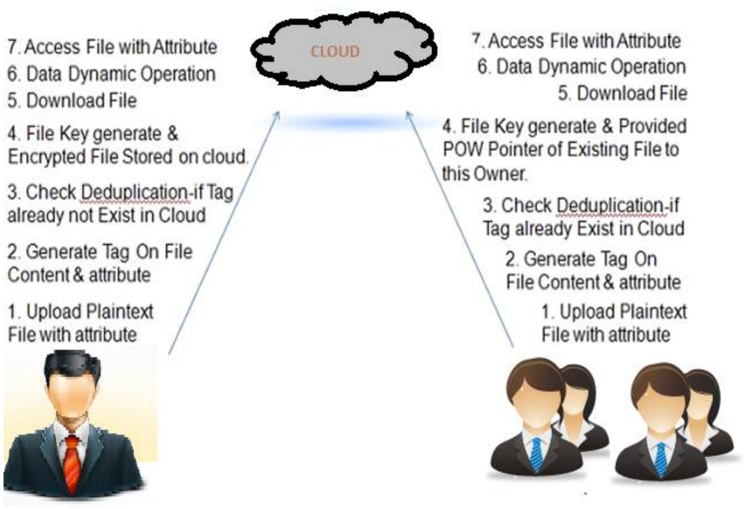
Vol. 5, Issue 1, January 2017

5] Title:Secure and Efficient Proof of Storage with Deduplication
Authors: Qingji Zheng, Shouhuai Xu
They shows the two aspects are able to coexist within the same framework. This is possible because of *The public verifiability offered by* PDP/POR *schemes can be naturally exploited to achieve* POW. This "one stone, two birds" phenomenon inspired them to propose the novel notion of Proof of Storage with Deduplication (POSD as well as guided them to design a concrete scheme that is provably secure in the Random Oracle model based on the Computational DiffieHellman (CDH) assumption.

III. PROPOSED SYSTEM

In this System model considers two types of entities: the cloud server and users, For each file, *original user* is the user who uploaded the file to the cloud server, while *subsequent user* is the user who proved the ownership of the file but did not actually upload the file to the cloud server.

System Architecture


CLOUD
7. Access File with Attribute
6. Data Dynamic Operation
5. Download File
4. File Key generate & Encrypted File Stored on cloud.
3. Check Deduplication-if Tag already not Exist in Cloud
2. Generate Tag On File Content & attribute
1. Upload Plaintext File with attribute
7. Access File with Attribute
6. Data Dynamic Operation
5. Download File
4. File Key generate & Provided POW Pointer of Existing File to this Owner.
3. Check Deduplication-if Tag already Exist in Cloud
2. Generate Tag On File Content & attribute
1. Upload Plaintext File with attribute

There are five phases in a deduplicatable dynamic PoS system:
1) pre-process
2) upload
3) deduplication
4) update
5) proof of storage

In the *pre-process* phase, users intend to upload their local files. The cloud server decides whether these files should be uploaded. If the upload process is granted, go into the upload phase; otherwise, go into the deduplication phase.

In the *upload* phase, the files to be uploaded do not exist in the cloud server. The original users encodes the local files and upload them to the cloud server.

In the *duplication* phase, the files to be uploaded already exist in the cloud server. The subsequent users possess the files locally and the cloud server stores the authenticated structures of the files. Subsequent users need to convince the cloud server that they own the files without uploading them to the cloud server.

Note that, these three phases (pre-process, upload, and deduplication) are executed only once in the life cycle of a file from the perspective of users. That is, these three phases appear only when users intend to upload files. If these phases terminate normally, i.e., users finish uploading in the upload phase, or they pass the verification in the deduplication phase, we say that the users have the ownerships of the files.

Some Advantages of Proposed System
1) The duplicate files are mapped with a single copy of the file by mapping with the existing file in the cloud
2) The comprehensive requirements in multi-user cloud storage systems and introduced the model of deduplicatable dynamic PoS.

IV. Calculation

There are total three users admin(cloud sever),user and subsequent user
User and subsequent user : Cloud Clients have large data les to be stored and rely on the cloud for data maintenance and computation. They can be either individual consumers or commercial organizations and they need security over data stored on cloud.

Admin : Cloud Servers virtualize the resources according to the requirements of clients and expose them as storage pools. Typically, the cloud clients may buy or lease storage capacity from cloud servers, and store their individual data in these bought or rented spaces for future utilization. also provides a proof of ownership to user for downloading of file.

V. Conclusion

We defined the comprehensive requirements in multi-user cloud storage systems and introduced the model of deduplicatable dynamic PoS with user previleges or attribute sharing .We used a tool HAT which is an efficient authenticated structure. By using HAT, we provide deduplicatable dynamic PoS scheme called DeyPoS and proved its security in the random oracle model. The analysis show that our DeyPoS implementation is efficient, especially when the file size and the number of the challenged blocks are large.

VI. Acknowledgement

We might want to thank the analysts and also distributers for making their assets accessible. We additionally appreciative to commentator for their significant recommendations furthermore thank the school powers for giving the obliged base and backing.

References

1. Z. Xia, X. Wang, X. Sun, and Q. Wang, “A Secure and Dynamic Multi- Keyword Ranked Search Scheme over Encrypted Cloud Data,” IEEE Transactions on Parallel and Distributed Systems, vol. 27, no. 2, pp. 340–352, 2016.
2. Z. Xiao and Y. Xiao, “Security and privacy in cloud computing,” IEEE Communications Surveys Tutorials, vol. 15, no. 2, pp. 843859, 2013.
3. C. A. Ardagna, R. Asal, E. Damiani, and Q. H. Vu, “From Security to Assurance in the Cloud: A Survey,” ACM Comput. Surv., vol. 48, no. 1, pp. 2:1–2:50, 2015.
4. 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, pp. 598–609, 2007.
5. G. Ateniese, R. Di Pietro, L. V. Mancini, and G. Tsudik, “Scalable and Efficient Provable Data Possession,” in Proc. Of SecureComm, pp. 1– 10, 2008. G. Ateniese, S. Kamara, and J. Katz, “Proofs of storage from homomorphic identification protocols,” in Proc. of ASIACRYPT, pp. 319–333, 2009.
6. C. Erway, A. K¨upc ¨u, C. Papamanthou, and R. Tamassia, “Dynamic provable data possession,” in Proc. of CCS, pp. 213–222, 2009.
7. R. Tamassia, “Authenticated Data Structures,” in Proc. of ESA, pp. 2–5, 2003.
8. 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, pp. 355–370, 2009.
9. F. Armknecht, J.-M. Bohli, G. O. Karame, Z. Liu, and C. A. Reuter, “Outsourced proofs of retrievability,” in Proc. of CCS, pp. 831–843, 2014.
10. H. Shacham and B. Waters, “Compact Proofs of Retrievability,” Journal of Cryptology, vol. 26, no. 3, pp. 442–483, 2013.
11. J. Douceur, A. Adya, W. Bolosky, P. Simon, and M. Theimer,“Reclaiming space from duplicate files in a serverless distributedfile system,” in Proc. of ICDCS, pp. 617–624, 2002.
12. A. Juels and B. S. Kaliski, Jr., “PORs: Proofs of retrievability forlarge files,” in Proc. of CCS, pp. 584–597, 2007.