RESEARCH ARTICLE OPEN ACCESS

DEYPOS: Deduplicatable Efficiency of Dynamic Proof of Storage for Multi-User Knowledge

[1] A. Senthil Kumar, [2] P.Raju
[1]Asst.professor, Dept.of.Computer Science, Tamil University, Thanjavur-613010.
[2]Research Scholar, Dept.of.Computer Science, Tamil University, Thanjavur-613010.

----------**********----------

Abstract:

Dynamic Proof of Storage (PoS) is a useful cryptographic primitive that enables a user to check the integrity of outsourced files and to efficiently update the files in a cloud server. Although researchers have proposed many dynamic PoS schemes in single user environments, the problem in multi-user environments has not been investigated sufficiently. A practical multi-user cloud storage system needs the secure client-side cross-user deduplication technique, which allows a user to skip the uploading process and obtain the ownership of the files immediately, when other owners of the same files have uploaded them to the cloud server. To the best of our knowledge, none of the existing dynamic PoSs can support this technique. In this paper, we introduce the concept of deduplicatable dynamic proof of storage and propose an efficient construction called DeyPoS, to achieve dynamic PoS and secure cross-user deduplication, simultaneously. Considering the challenges of structure diversity and private tag generation, we exploit a novel tool called Homomorphic Authenticated Tree (HAT). We prove the security of our construction, and the theoretical analysis and experimental results show that our construction is efficient in practice.

Keywords— **Authenticate, Techniques, Tree.**

----------**********----------

I. INTRODUCTION

To the best of our knowledge, none of the existing dynamic PoSs can support this technique. In this paper, we introduce the concept of deduplicatable dynamic proof of storage and propose an efficient construction called DeyPoS, to achieve dynamic PoS and secure cross-user deduplication, simultaneously. Considering the challenges of structure diversity and private tag generation, we exploit a novel tool called Homomorphic Authenticated Tree (HAT). We prove the security of our construction, and the theoretical analysis and experimental results show that our construction is efficient in practice.

To better understand the following contents, we present more details about PoS and dynamic PoS. In these schemes, each block of a file is attached a tag which is used for verifying the integrity of that block. When a verifier wants to check the integrity of a file, it randomly selects some block indexes of the file, and sends them to the cloud server. According to these challenged indexes, the cloud server returns the corresponding blocks along with their tags. The verifier checks the block integrity and index correctness. The former can be directly guaranteed by cryptographic tags. How to deal with the latter is the major difference between PoS and dynamic PoS.

In most of the PoS schemes, the block index is “encoded” into its tag, which means the verifier can check the block integrity and index correctness simultaneously. However, dynamic PoS cannot encode the block indexes into tags,

since the dynamic operations may change many indexes of non-updated blocks, which incurs unnecessary computation and communication cost. For example, there is a file consisting of 1000 blocks, and a new block is inserted behind the second block of the file. Then, 998 block indexes of the original file are changed, which means the user has to generate and send 999 tags for this update. Authenticated structures are introduced in dynamic PoSs to solve this challenge. As a result, the tags are attached to the authenticated structure rather than the block indexes.

Taking the Merkle tree in Fig. 1a as an example Merkle tree is one of the most efficient authenticated structures in dynamic PoS, the tag corresponding to the second file block involves the index of the Merkle tree node v5, that is 5, rather than 2. When a new block is inserted behind the second file block, the authenticated structure turns into the structure. Then, the index in the tag corresponding to the second file block changes, and the user only has to generate 2 tags for this update. This figure provides an instance that authenticated structure used in dynamic PoS reduces the computation cost in the update process.

Taking the combination of as example, is a dynamic PoS scheme which employs Merkle tree as its authenticated structure, is a crossuser deduplication scheme which also employs Merkle tree as its authenticated structure. Suppose Alice and Bob independently own a file F, a Merkle tree TF is generated and stored by the cloud server for deduplication, and two Merkle trees TA and TB are generated by Alice and Bob respectively, and stored in the cloud server for PoS. When Alice updates F to F′, the cloud server updates TA to T′A for PoS and generates a new Merkle tree TF′ for deduplication.

EXISTING OF THREATS:

Storage outsourcing is becoming more and more attractive to both industry and academia due to the advantages of low cost, high accessibility, and easy sharing. As one of the storage outsourcing forms, cloud storage gains wide attention in recent years. Many companies, provide their own cloud storage services, where users can upload their files to the servers, access them from various devices, and share them with the others. Although cloud storage services are widely adopted in current days, there still remain many security issues and potential threats.

Data integrity is one of the most important properties when a user outsources its files to cloud storage. Users should be convinced that the files stored in the server are not tampered. Traditional techniques for protecting data integrity, such as message authentication codes (MACs) and digital signatures, require users to download all of the files from the cloud server for verification, which incurs a heavy communication cost. These techniques are not suitable for cloud storage services where users may check the integrity frequently, such as every hour.

DISADVANTAGES

- Existing techniques are not suitable for cloud storage services.
- Frequently users may check the integrity.

PROPOSED FOR ALGORITHMS:

We propose a concrete scheme of deduplicatable dynamic PoS called DeyPoS. It consists of five algorithms.

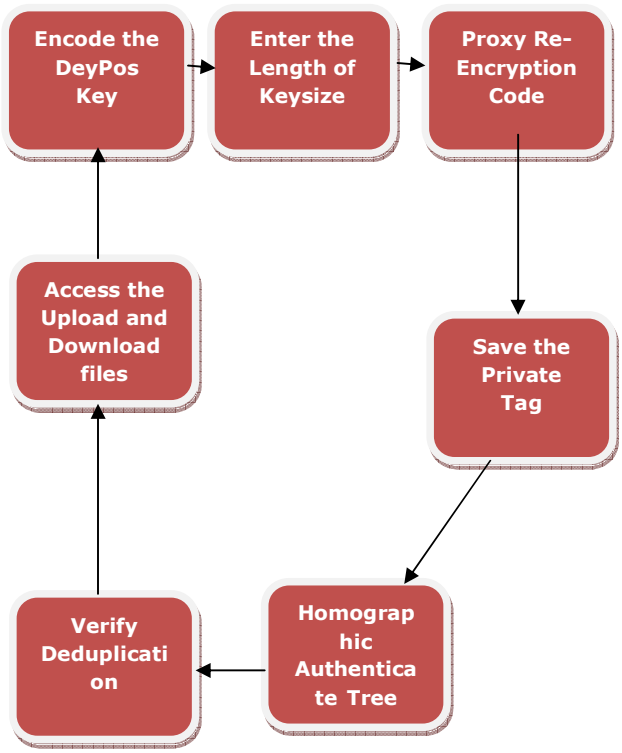
- Init
- Encode
- Deduplicate
- Update
- Check.

Our system model considers two types of entities: the cloud server and users, as shown in Fig. 2. 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.

ADVANTAGES

- DeyPoS techniques are suitable for cloud storage services.
- Doesn't need to frequently users may check the integrity.



Encode And Deduplicate

PROCESS KEYWORDS:

1. **Init**
   Cloud Server and user register the Unique ID for initialization. Original registered user can upload the files to the server. Subsequent user register the Unique ID and its registered Password for access the uploaded files.
2. **Encode**
   Original users before upload the Files to the Cloud server a encoding process done. In the Encode process the Homographic Authenticate Tree logic be applied.
3. **Deduplicate**
   Detect the duplicate of the ID by verify the database by the Unique Deypos ID and the generated unique password. If ID and password validated success the subsequent users can access the file rights otherwise ID consider as Duplication
4. **Update**
   Original users upload the file to the cloud server and then updated. File upload with unique ID for access the files by the subsequent users.
5. **Check.**
   Check the Validation and verification process for the Files upload and download. Cloud server performance and No. of deduplication trials happen when try to access the server files.

V. CONCLUSION

We proposed the comprehensive requirements in multi-user cloud storage systems and introduced the model of deduplicatable dynamic Pos. We designed a novel tool called HAT which is an efficient authenticated structure. Based on HAT, we proposed the first practical deduplicatable dynamic PoS scheme called DeyPoS and proved its security in the random oracle model. The theoretical and experimental results show that our DeyPos implementation is efficient, especially when the file size and the number of the challenged blocks are large. The first realistic deduplicatable dynamic PoS scheme which makes use of complete necessities in multi-consumer cloud storage systems and proved its security within the random oracle model. The theoretical and experimental results show that the procedure is efficient, peculiarly

when the file dimension and the number of the challenged blocks are large.

VI.REFERENCE

[1] S. Kamara and K. Lauter, "Cryptographic cloud storage," in *Proc. of FC*, pp. 136–149, 2010.

[2] 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.

[3] Z. Xiao and Y. Xiao, "Security and privacy in cloud computing," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 843–859, 2013.

[4] 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.

[5] 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.

[6] 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.

[7] G. Ateniese, S. Kamara, and J. Katz, "Proofs of storage from homomorphic identification protocols," in *Proc. of ASIACRYPT*, pp. 319–333, 2009.

[8] C. Erway, A. K¨upc ¨u, C. Papamanthou, and R. Tamassia, "Dynamic provable data possession," in *Proc. of CCS*, pp. 213–222, 2009.

[9] R. Tamassia, "Authenticated Data Structures," in *Proc. of ESA*, pp. 2–5, 2003.

[10] 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.