VERIDEDUP: ENSURING DATA INTEGRITY AND PROOF OF DUPLICATION IN CLOUD-BASED DEDUPLICATION SYSTEMS

K.PUNYAVATHI,M.Tech Student
Mr. K.SAMSON PAUL, Assistant Professor
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
Dr. K. V. SUBBA REDDY INSTITUTE OF TECHNOLOGY, OPP.DUPADU (RS), NH-44, LAKSHMIPURAM (PO), KURNOOL-518218.

Abstract—Data deduplication is a technique to eliminate duplicate data in order to save storage space and enlarge upload bandwidth, which has been applied by cloud storage systems. However, a cloud storage provider (CSP) may tamper user data or cheat users to pay unused storage for duplicate data that are only stored once. Although previous solutions adopt message-locked encryption along with Proof of Retrievability (PoR) to check the integrity of deduplicated encrypted data, they ignore proving the correctness of duplication check during data upload and require the same file to be derived into same verification tags, which suffers from brute-force attacks and restricts users from flexibly creating their own individual verification tags. In this paper, we propose a verifiable deduplication scheme called VeriDedup to address the above problems. It can guarantee the correctness of duplication check and support flexible tag generation for integrity check over encrypted data deduplication in an integrative way. Concretely, we propose a novel Tag-flexible Deduplication-supported Integrity Check Protocol (TDICP) based on Private Information Retrieval (PIR) by introducing a novel verification tag called note set, which allows multiple users holding the same file to generate their individual verification tags and still supports tag deduplication at the CSP. Furthermore, we make the first attempt to guarantee the correctness of data duplication check by introducing a novel User Determined Duplication Check Protocol (UDDCP) based on Private Set Intersection (PSI), which can resist a CSP from providing a fake duplication check result to users. Security analysis shows the correctness and soundness of our scheme. Simulation studies based on real data show the efficacy and efficiency of our proposed scheme and its significant advantages over prior arts.

I.INTRODUCTION

CLOUD computing has become a popular information technology service by providing huge amount of resources (e.g., storage and computing) to end users based on their demands. Among all cloud computing services, cloud storage is the most popular. Since the volume of data in the world is increasing rapidly, saving cloud storage becomes essential. One of the key reasons that causes storage waste is duplicate data storage. Multiple users may save same files or different files containing same pieces of data blocks at the cloud. Obviously, duplicate data storage at the cloud introduces a big waste of storage resources. Data de duplication [1]–[3] provides a promising solution to this issue. In a de duplication scheme, the CSP can cooperate with the cloud user to first check whether a pending uploaded file has been saved already or not, and then provide the user whose pieces of file data are checked duplicate a way to access the file without storing another copy at the cloud.

However, since the CSP cannot be fully trusted, the cloud users may suffer from some security and privacy issues. Notably, a semi trusted CSP may modify, tamper or delete the uploaded data driven by some profits. The damage of de duplicated data could cause huge loss to all related users (e.g., data owners and holders). Thus, the integrity of the data stored at the cloud should be verified, especially for duplicate data storage with de duplication.

Several Proof of Retriev ability (POR) schemes [4]–[9] have been proposed to address the issue of integrity check on cloud data storage in recent decade. In such schemes, a user adds verification tags along with a file. During the verification, the user creates a random challenge and sends it to the CSP, the CSP has to use all the data in user's corresponding files it stored as inputs to compute a response back to the user. The user then checks the integrity of the stored file by verifying the response. However, existing POR solutions mainly aim to improve the performance at the user side and assume that the CSP has infinite computation and storage resources. While, in practice, the CSP performs data de duplication in order to achieve the most economic usage of its storage. Unfortunately, existing solutions mentioned above are incompatible with de duplication. This is because the verification tags of these schemes are created with user individual private keys unknown to each other, thus different verification tags are generated, given the same file held by different users. But these verification tags cannot be de duplicated at the CSP as shown in Fig. 1(a).

Message-locked POR [10], [11] provides a promising solution to check data integrity when performing de duplication. It derives a same file into a same verification tag based on message-locked encryption technique as shown in Fig. 1(b). However, such design restricts the users from creating their own individual tags with their private keys. Practically, we expect an effective method that can check data integrity with the support of de duplication where each user can generate its own individual verification tags from its private key against brute-force attacks.

Another security issue ignored by the previous literature is the correctness guarantee of data duplication check provided by the CSP. Several schemes [12], [13] motivate the CSP to perform de duplication, but ignore that the CSP could cheat the users by providing a fake duplication check result. The reason is simple
since the CSP can gain an extra profit by asking the users to pay normal storage fee without granting a deserved discount while performing de duplication to save storage space. As shown in Table 1, we illustrate four situations that the CSP deals with a duplication check about file storage. We find that a problem may happen in the third situation that the CSP actually has the file tested duplicate but tells the user that it is not in order to let the user pay a normal storage fee without any discount, which should be offered due to de duplication and storage saving. By saving extra storage space, the CSP can earn more by serving more users with the same dishonest way. An effective mechanism should be proposed to prevent the user from being cheated by the CSP in the phase of data duplication check.

In this paper, we propose a novel de duplication scheme called VeriDedup to tackle the above two security issues in an integrative way. It contains a novel Tag-flexible De duplication-supported Integrity Check Protocol (TDICP) and a novel User Determined Duplication Check Protocol (UDDCP). The TDICP explores a new verification tag called note set in which each note is a randomized bit sequence that is conform to a function f. The note set is inserted into the files based on Private Information Retrieval (PIR). TDICP allows the users to create their own individual verification tags to check data integrity over the CSP with de duplication compatibility. Meanwhile, the UDDCP explores a new challenge and response mechanism based on Private Set Intersection (PSI) to let the user instead of the CSP tell whether the file is duplicate first, so that the CSP cannot cheat the user on the result of duplication check during file upload. VeriDedup is built upon our previous scheme [14], which offers such functionalities as de duplication over cipher text, Proof of Ownership (POW) and key assignment by employing proxy re-encryption (PRE). While, in this paper, we focus on integrity check and duplication check that are ignored in [14]. Thus, we assumed the functionalities of POW and encrypted data de duplication are available and are not the focus of this paper.

Specifically, the main contributes of this paper are summarized as below:

_ We propose a novel protocol named TDICP based on PIR to check the integrity of uploaded files in the CSP with de duplication employed. TDICP allows users to generate their own individual verification tags for integrity check while the verification tags can also be de duplicated at the CSP although different.

_ We propose another novel protocol named UDDCP to guarantee the correctness of duplication check based on PSI, so that the CSP is impossible to cheat the user to pay for unused storage space due to de duplication.

_ We construct a novel de duplication scheme called VeriDedup that contains the above two novel protocols and other essential properties, such as POW and data access key assignment by re-shaping our previous scheme in [14] in order to overcome its shortcomings regarding integrity and duplication proof.

_ We prove the security of TDICP and UDDCP by constructing several games and conduct both theoretical analysis and experimental simulation to evaluate their performance. Our results show their efficacy and efficiency.

II. LITERATURE SURVEY

"Heterogeneous data storage management with deduplication in cloud computing,

Z. Yan, L. Zhang, W. Ding, and Q. Zheng,

Cloud storage as one of the most important services of cloud computing helps cloud users break the bottleneck of restricted resources and expand their storage without upgrading their devices. In order to guarantee the security and privacy of cloud users, data are always outsourced in an encrypted form. However, encrypted data could incur much waste of cloud storage and complicate data sharing among authorized users. We are still facing challenges on encrypted data storage and management with deduplication. Traditional deduplication schemes always focus on specific application scenarios, in which the deduplication is completely controlled by either data owners or cloud servers. They cannot flexibly satisfy various demands of data owners according to the level of data sensitivity. In this paper, we propose a heterogeneous data storage management scheme, which flexibly offers both deduplication management and access control at the same time across multiple Cloud Service Providers (CSPs). We evaluate its performance with security analysis, comparison and implementation. The results show its security, effectiveness and efficiency towards potential practical usage.

"A scheme to manage encrypted data storage with deduplication in cloud,"

Z. Yan, W. X. Ding, and H. Q. Zhu,

Cloud computing plays an important role in supporting data storage, processing, and management in the Internet of Things (IoT). To preserve cloud data confidentiality and user privacy, cloud data are often stored in an encrypted form. However, duplicated data that are encrypted under different encryption schemes could be stored in the cloud, which greatly decreases the utilization rate of storage resources, especially for big data. Several data deduplication schemes have recently been proposed. However, most of them suffer from security weakness and lack of flexibility to support secure data access control. Therefore, few can be deployed in practice. This article proposes a scheme based on attribute-based encryption (ABE) to deduplicate encrypted data stored in the cloud while also supporting secure data access control. The authors evaluate the scheme's performance based on analysis and implementation. Results show the efficiency, effectiveness, and scalability of the scheme for potential practical deployment.

"Lightweight cloud storage auditing with deduplication supporting strong privacy protection,"

W. Shen, Y. Su, and R. Hao,

The cloud storage auditing with deduplication is able to verify the integrity of data stored in the cloud while the cloud needs to keep only a single copy of duplicated file. To the best of our knowledge, all of the existing cloud storage auditing schemes with deduplication are vulnerable to brute-force dictionary attacks, which incurs the leakage of user privacy. In this paper, we focus on a new aspect of being against brute-force dictionary attacks on cloud storage auditing. We propose a cloud storage auditing scheme with deduplication supporting strong privacy protection, in which the privacy of user's file would not be disclosed to the cloud and other parties when this user's file is predictable or from a small space. In the proposed scheme, we design a novel method to generate the file index for duplicate check, and use a new strategy to generate the key for file encryption.

In addition, the user only needs to perform lightweight computation to generate data authenticators, verify cloud data integrity, and retrieve the file from the cloud. The security proof and the performance evaluation demonstrate that the proposed scheme achieves desirable security and efficiency.

"Secure and efficient proof of storage with deduplication,"

Q. Zheng and S. Xu,

Both security and efficiency are crucial to the success of cloud storage. So far, security and efficiency of cloud storage have been separately investigated as follows: On one hand, security notions such as Proof of Data Possession (PDP) and Proof of Retrievability (POR) have been introduced for detecting the tamperation of data stored in the cloud. One the other hand, the notion of Proof of Ownership (POW) has also been proposed to alleviate the cloud server from storing multiple copies of the same data, which could substantially reduce the consumption of both network bandwidth and server storage space. These two aspects are seemingly quite to the opposite of each other. In this paper, we show, somewhat surprisingly, that the two aspects can actually coexist within the same framework. This is possible fundamentally because of the following insight: The public verifiability offered by PDP/POR schemes can be naturally exploited to achieve POW. This "one stone, two birds" phenomenon not only inspired us to propose the novel notion of Proof of Storage with Deduplication (POSD), but also guided us to design a concrete scheme that is provably secure in the Random Oracle model based on the Computational DiffieHellman (CDH) assumption.

III. SYSTEM ANALYSIS

EXISTINGSYSTEM

Shacham and Waters [18] proposed a new solution based on their proposed concept of Compact PoR, which adopts an erasure code and an authenticator with a BLS signature [19] and Message Authentication Codes (MAC) [11]. However, the computational complexity of generating the authenticator is high and the number of the authenticators is linear to the number of blocks. Xu and Chang [16] proposed to enhance the scheme in [18] with an polynomial commitment [18] to reduce communication cost. Azraoui et al. [20] proposed a scheme called Stealth Guard by using PIR within Word Search (WS) technique to retrieve a witness of watchdogs (similar as tags) and allows an unlimited number of queries. Compared with other works, the generation of watchdogs is more lightweight than the generation of tags like in [7], [18]. In addition, the overhead of storing the watchdogs is less than that of previous work. However, those works fail in supporting deduplication over verification tags.

Zheng et al. [5] introduced a new proof of storage scheme with deduplication based on a publicly verifiable proof of data possession. In their scheme, users can verify the correct storage of deduplicated data with the key of the first user who actually uploads the file. However, this scheme has been proved insecure under a weak key attack in [25] and it cannot prevent the users from being cheated by the CSP. Vassilopoulos et al. [10] proposed a scheme by transforming the existing PoR into a form that is message-locked and integrating it with a deduplication function. However, these works require to derive the same file into the same verification tag. But multiple users holding the same file stored at the cloud may create different tags as their willingness for data integrity check, which improves integrity check security by overcoming brute-force attacks, but impacts deduplication.

Disadvantages

An existing system can't give solutions for the following issues.

1) Snooping the private data of the data holders;

2) Cheating the data holders by providing a wrong duplication check result in order to ask a higher storage fee;

3) Causing data loss due to carelessness of data maintenance

Proposed System

❖ The system proposes a novel protocol named TDICP based on PIR to check the integrity of uploaded files in the

CSP with deduplication employed. TDICP allows users to generate their own individual verification tags for integrity check while the verification tags can also be deduplicated at the CSP although different.

- The system proposes another novel protocol named UDDCP to guarantee the correctness of duplication check based on PSI, so that the CSP is impossible to cheat the user to pay for unused storage space due to deduplication.
- The system constructs a novel deduplication scheme called VeriDedup that contains the above two novel protocols and other essential properties, such as PoW and data access key assignment by re-shaping our previous scheme in [14] in order to overcome its shortcomings regarding integrity and duplication proof.
- The system proves the security of TDICP and UDDCP by constructing several games and conduct both theoretical analysis and experimental simulation to evaluate their performance. Our results show their efficacy and and efficiency.

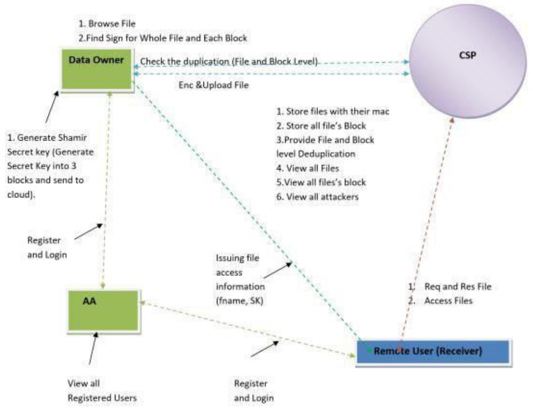
Advantages

Independent integrity check when deduplication: VeriDedup allows the data holder to check the integrity of its files stored at the CSP without downloading the whole files and interacting with the corresponding data owner.

Flexible tag generation: VeriDedup allows each data holder to create its own individual verification tags while still can perform data deduplication over those tags.

Correctness guarantee of duplication check: VeriDedup can assure the correctness of duplication check. Thus, a semi-trusted CSP can never cheat the data holders to upload any data that have already been stored by the CSP.

IV.ARCHITECTURE DIAGRAM


1. Browse File
2.Find Sign for Whole File and Each Block
Data Owner
Check the duplication (File and Block Level)
Enc &Upload File
CSP
1. Store files with their mac
2. Store all file's Block
3.Provide File and Block level Deduplication
4. View all Files
5.View all files's block
6. View all attackers
1. Generate Shamir Secret key (Generate Secret Key into 3 blocks and send to cloud).
Register and Login
Issuing file access information (fname, SK)
1. Req and Res File
2. Access Files
AA
Remote User (Receiver)
View all Registered Users
Register and Login

V. MPLEMENTATION

Data Owner

In this module, the data owner uploads their data in the cloud server. For the security purpose the data owner encrypts the data file and then store in the cloud. The data owner can check the duplication of the file over Corresponding cloud server. The Data owner can have capable of manipulating the encrypted data file and the data owner can check the multiple cloud data as well as the duplication of the specific file. And also he can create remote user with respect to registered cloud servers. And also data owner has migrate to another cloud option, by this he can migrate files from one cloud server to another cloud server.

AA

In this module, the connector helps to check duplication of file existed or not in cloud server and you can check in multi cloud servers also. If it is existed then also owner trying to upload the same file in same cloud server then connector automatically blocks his access permission. If it is not existed then data owner can upload file in multi cloud servers at a time.

Cloud Server

The cloud service provider manages a cloud to provide data storage service. Data owners encrypt their data files and store them in the cloud for sharing with Remote User. To access the shared data files, data consumers download encrypted data files of their interest from the cloud and then decrypt them

Remote User

In this module, remote user logs in by using his user name and password. After he will request for secrete key of required file from cloud servers, and get the secrete key. After getting secrete key he is trying to download file by entering file name and secrete key from cloud server.

Attacker Module

In remote user module, while downloading time if remote user entered any wrong file name or secrete key then cloud servers treats him as attacker and moves his access permission to block/attacker li

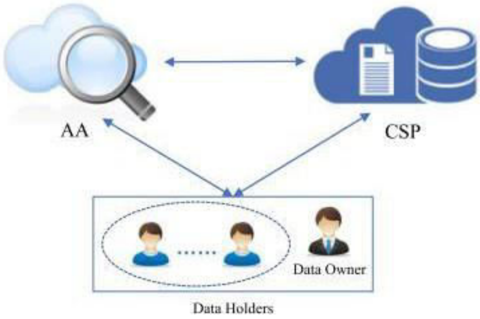
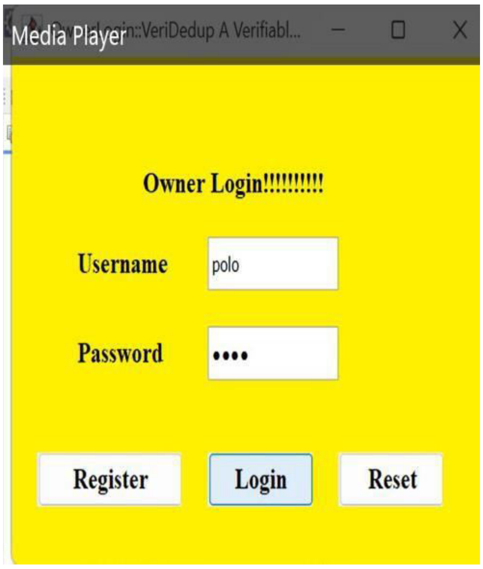


Fig: System model

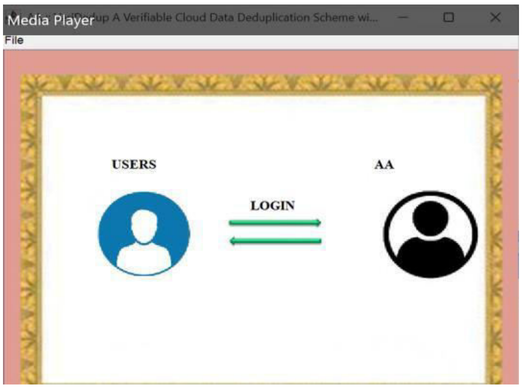
VI. SCREENSHORTS

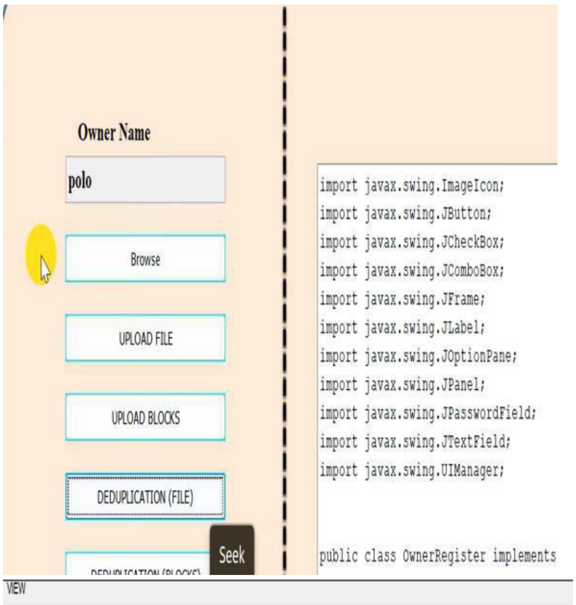


Owner Login!!!!!!!!!

Username polo

Password ••••

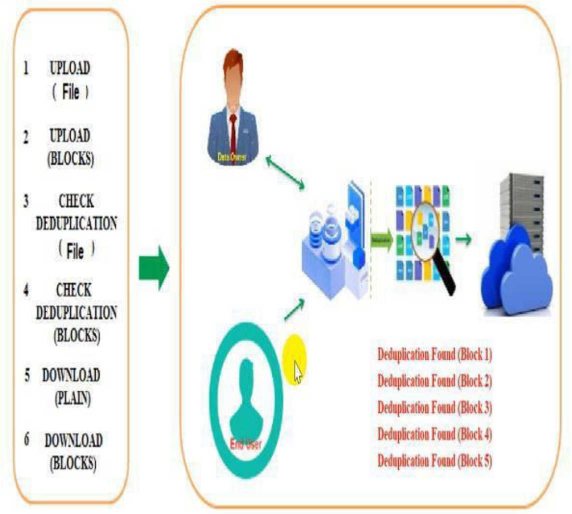

Media Player
VeriDedup A Verifiable Cloud Data Deduplication Scheme wi...
File
USERS
LOGIN
AA


Owner Name

polo

```
import javax.swing.ImageIcon;
import javax.swing.JButton;
import javax.swing.JCheckBox;
import javax.swing.JComboBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JPasswordField;
import javax.swing.JTextField;
import javax.swing.UIManager;


public class OwnerRegister implements
```


1 UPLOAD (File)

2 UPLOAD (BLOCKS)

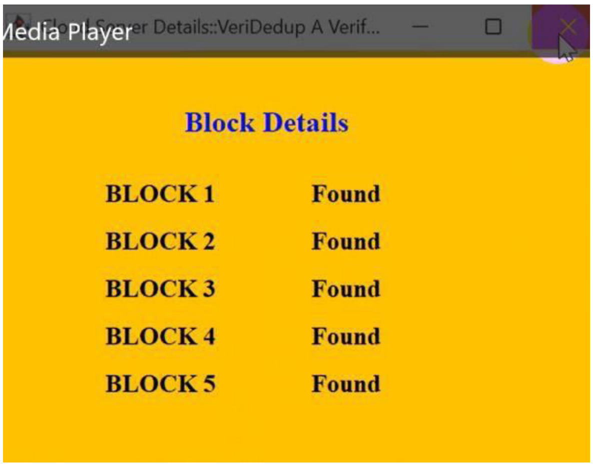
3 CHECK DEDUPLICATION (File)

4 CHECK DEDUPLICATION (BLOCKS)

5 DOWNLOAD (PLAIN)

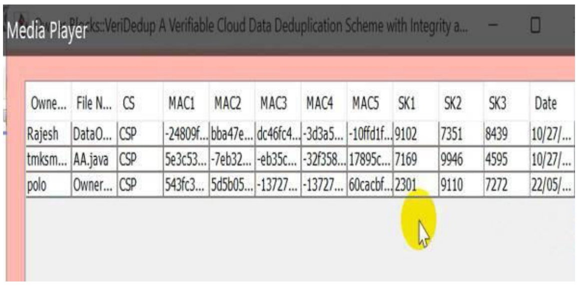
6 DOWNLOAD (BLOCKS)

Deduplication Found (Block 1)
Deduplication Found (Block 2)
Deduplication Found (Block 3)
Deduplication Found (Block 4)
Deduplication Found (Block 5)

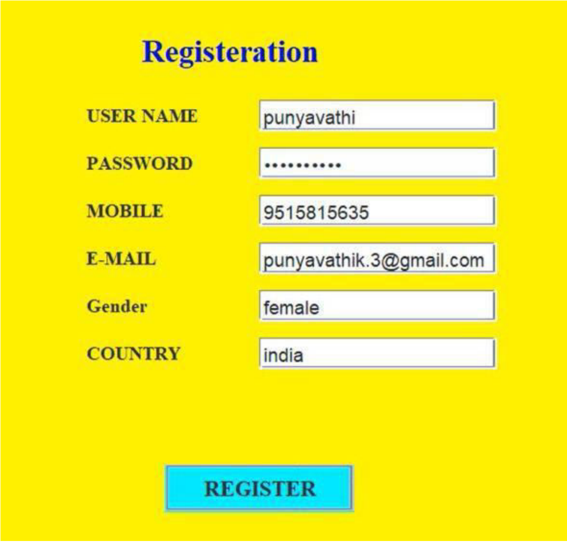


Block Details

BLOCK 1	Found
BLOCK 2	Found
BLOCK 3	Found
BLOCK 4	Found
BLOCK 5	Found

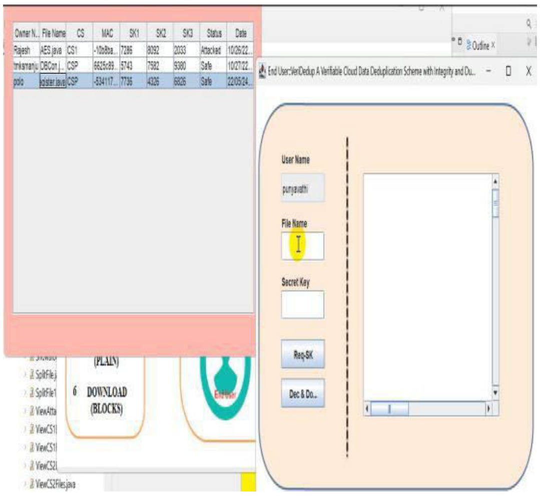

Media Player
...Blocks::VeriDedup A Verifiable Cloud Data Deduplication Scheme with Integrity a...

Owne...	File N...	CS	MAC1	MAC2	MAC3	MAC4	MAC5	SK1	SK2	SK3	Date
Rajesh	DataO...	CSP	-24809f...	bba47e...	dc46fc4...	-3d3a5...	-10ffd1f...	9102	7351	8439	10/27/...
tmksm...	AA.java	CSP	5e3c53...	-7eb32...	-eb35c...	-32f358...	17895c...	7169	9946	4595	10/27/...
polo	Owner...	CSP	543fc3...	5d5b05...	-13727...	-13727...	60cacbf...	2301	9110	7272	22/05/...



Registeration

USER NAME	punyavathi
PASSWORD	••••••••••
MOBILE	9515815635
E-MAIL	punyavathik.3@gmail.com
Gender	female
COUNTRY	india



Owner N...	File Name	CS	MAC	SK1	SK2	SK3	Status	Date
Rajesh	AES.java	CS1	-10b8ba...	7286	8092	2033	Attacked	10/26/22...
mksmanju	DBCon.j...	CSP	6625c89...	5743	7582	9380	Safe	10/27/22...
polo	egister.java	CSP	-534117...	7736	4326	6826	Safe	22/05/24...

End User:VeriDedup A Verifiable Cloud Data Deduplication Scheme with Integrity and Du...

User Name: punyavathi

VII. CONCLUSION

In this paper, we introduced VeriDedup to check the integrity of an outsourced encrypted file and guarantee the correctness of duplication check in an integrated way. The integrity check protocol TDICP of VeriDedup allows multiple data holders to verify the integrity of their outsourced file with their own individual verification tags without interacting with the data owner. On the other hand, we employed a novel challenge and response mechanism in the duplication check protocol UDDCP of VeriDedup to let the data holder instead of the CSP first tell whether a file is duplicate in order to guarantee the correctness of duplication check. Security and performance analysis show that VeriDedup is secure and efficient under the described security model. The result of our computer simulation further shows its efficiency compared with highly related prior arts.

REFERENCES

[1] Z. Yan, L. F. Zhang, W. X. Ding, and Q. H. Zheng, “Heterogeneous data storage management with deduplication in cloud computing,” IEEE Transactions on Big Data, pp. 1–1, 2017.

[2] Z. Yan, W. X. Ding, and H. Q. Zhu, “A scheme to manage encrypteddata storage with deduplication in cloud,” in International Conference on Algorithms and Architectures for Parallel Processing, 2015.

[3] Z. Yan, M. J. Wang, Y. X. Li, and A. V. Vasilakos, “Encrypted data management with deduplication in cloud computing,” IEEE Cloud Computing, vol. 3, no. 2, pp. 28–35, 2016.

[4] W. Shen, Y. Su, and R. Hao, “Lightweight cloud storage auditing with deduplication supporting strong privacy protection,” IEEE Access, vol. 8,pp. 44 359–44 372, 2020.

[5] Q. Zheng and S. Xu, “Secure and efficient proof of storage with deduplication,” in CODASPY ’12, New York, NY, USA, 2012, p. 1–12.

[6] A. Giuseppe, R. Burns, and C. Reza, “Provable data possession at untrusted stores,” in Proceedings of the 14th ACM Conference on Computer and Communications Security, 2007, pp. 598–609.

[7] G. Ateniese, R. Burns, R. Curtmola, J. Herring, O. Khan, Z. Peterson, and D. Song, “Remote data checking using provable data possession,” ACM Transactions on Information and System Security, vol. 14, pp. 1–34, 2011.

[8] Z. Wen, J. Luo, H. Chen, J. Meng, X. Li, and J. Li, “A verifiable data deduplication scheme in cloud computing,” in INCOS ’14, USA, 2014,p. 85–90.

[9] P. Meye, P. Raïpin, F. Tronel, and E. Anceaume, “A secure two-phase data deduplication scheme,” in HPCC ’14, CSS ’14, ICESS ’14, 2014,pp. 802–809.

[10] D. Vasilopoulos, M. Önen, K. Elkhiyaoui, and R. Molva, “Messagelocked proofs of retrievability with secure deduplication,” in Proceedings of the 2016 ACM on Cloud Computing Security Workshop, 2016, pp. 73–83.

[11] M. Bellare, R. Canetti, and H. Krawczyk, "Keying hash functions for message authentication," in CRYPTO '96, Berlin, Heidelberg, 1996, pp.1–15.

[12] X. Q. Liang, Z. Yan, X. F. Chen, L. T. Yang, W. J. Lou, and Y. T. Hou,"Game theoretical analysis on encrypted cloud data deduplication," IEEETransactions on Industrial Informatics, vol. 15, no. 10, pp. 5778–5789,

2019. [13] X. Q. Liang, Z. Yan, R. H. Deng, and Q. H. Zheng, "Investigating theadoption of hybrid encrypted cloud data deduplication with game theory,"

IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 3,pp. 587–600, 2021.

[14] Z. Yan, W. Ding, X. Yu, H. Zhu, and R. H. Deng, "Deduplication onencrypted big data in cloud," IEEE Transactions on Big Data, vol. 2,no. 2, pp. 138–150, 2016.