International Conference on Science, Engineering
& Management Trends

International Journal of Science,
Engineering and Technology
ISSN: 2348-4098, P-ISSN: 2395-4752

A Secure and Transparent Deduplication Scheme for Cloud Storage Integrity

A. Hari Hara Sumanth Sai, N.Leela Mohan Sreenivas, Vasu R, Rishitha S, Dr.S.Lavanya
C.S.E
Muthayammal Engineering College Rasipuram, Tamilnadu

Abstract- One method that cloud storage systems use to reduce space and increase upload bandwidth is data deduplication, which involves removing duplicate data. On the other hand, a cloud storage provider (CSP) might alter user data or trick consumers into paying for storage space that isn't being utilized for data that is duplicated. While there have been solutions that use message-locked encryption and Proof of Retrievability (PoR) to ensure deduplicated encrypted data is intact, these solutions fail to prove that the duplication check is correct when data is uploaded. Additionally, they force users to use the same file for verification tags, which leaves them vulnerable to brute-force attacks and limits their ability to create unique verification tags. We provide VeriDedup, a verified deduplication technique, in this work to solve the aforementioned issues. With its integrated support for configurable tag creation for integrity checks over encrypted data deduplication, it can ensure that duplication checks are accurate. To be more specific, we suggest a new protocol called TDICP that is built on Private Information Retrieval (PIR). This protocol introduces a new verification tag called note set that enables several users to create their own verification tags while still supporting tag deduplication at the CSP.

Keywords- Data deduplication, Data security

I. INTRODUCTION

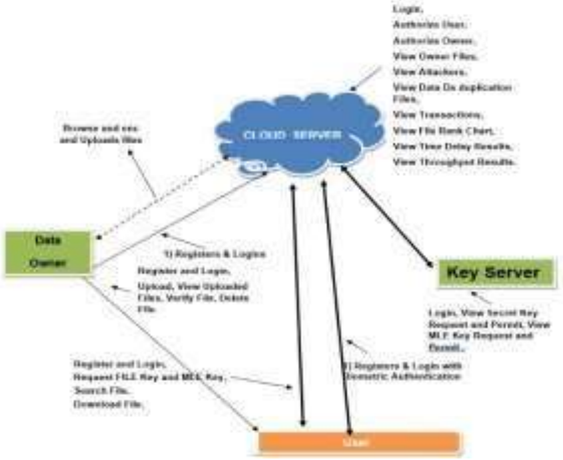
Cloud storage systems often use data deduplication to maximize upload bandwidth and storage space by detecting and removing duplicate data. Despite data deduplication's many advantages, worries about data integrity and user confidence have arisen with its widespread usage. Specifically, there is a risk of cloud storage providers (CSPs) exploiting deduplication processes for financial gain, potentially compromising user data integrity and privacy. Prior approaches to address these concerns have primarily focused on encryption and Proof of Retrievability (PoR) mechanisms, yet they have often overlooked the critical aspect of verifying the correctness of deduplication checks during data upload.

In response to these challenges, this paper presents VeriDedup, an innovative verifiable deduplication scheme tailored to ensure the integrity of data deduplication processes within cloud storage infrastructures. Two important protocols are introduced by VeriDedup: UDDCP and TDICP, which stand for Tagflexible Deduplication-supported Integrity Check Protocol. TDICP allows for customizable tag creation for integrity checks over encrypted data by using Private Information Retrieval (PIR) methods. This paper introduces a new confirmation label called the note set that permits numerous clients to make their own check labels while as yet permitting label deduplication at the CSP level.

This approach not only enhances flexibility for users but also strengthens data integrity checks within the cloud storage environment. In addition, the article presents UDDCP, a method for validating data duplication checks that relies on Private Set Intersection (PSI) concepts. By employing UDDCP, VeriDedup mitigates the risk of CSPs providing false duplication check results to users, thereby bolstering trust and confidence in cloud storage services.

Simulation tests using real-world data show how effective and efficient VeriDedup is in comparison to current solutions, and security study on VeriDedup supports its accuracy and resilience. Overall, VeriDedup represents a significant advancement in addressing critical concerns surrounding data integrity, trust, and efficiency in cloud storage deduplication processes, offering a comprehensive and reliable solution for users and organizations leveraging cloud storage services.

II. SYSTEM ARCHITECTURE



III. METHODOLOGY

Correctness of TDICP: We first prove the correctness of TDICP on extracting the queried column where the notes are inserted based on the PIR algorithm. During the Integrity check phase, the data holder computes as follows:

$$\begin{aligned} Resp * b^{-1} \bmod m &= (v \times D) * b^{-1} \bmod m \\ &= (be \times D) * b^{-1} \bmod m \\ &= e \times D \bmod m \end{aligned}$$

Since $e = (e_1, \ldots, e_t)$ and $t = x$,

$$D = \begin{pmatrix} d_{11} & \cdots & d_{1y} \\ \vdots & \ddots & \vdots \\ d_{x1} & \cdots & d_{xy} \end{pmatrix}$$

then

$$\begin{aligned} e \times D \bmod m &= \left(\sum_{i=1}^{x} e_i d_{i1}, \sum_{i=1}^{x} e_i d_{i2}, \ldots, \sum_{i=1}^{x} e_i d_{iy} \right) \bmod m \\ &= \left(\sum_{i=1}^{x} e_i d_{i1}, \sum_{i=1}^{x} e_i d_{i2}, \ldots, \sum_{i=1}^{x} e_i d_{iy} \right) \end{aligned}$$

When e_i is the queried column, $e_i = N^l + a_1 N^r$,

We have $\sum xi{=}1\, eidij = \sum_{i=1}^{x}(N^{l} + a_1N^{r})\, dij$,
then $\sum xi{=}1\, eidij \bmod Nr = \sum xi{=}1\, Nldij$
Otherwise, $e_i = a_kN^r$, we have $\sum_{i=1}^{x} e_id_{ij} =$
$\sum_{i=1}^{x}(a_kN^r)\, d_{ij}$, then $\sum^{x}{}_{i=1}\, e_id_{ij} \bmod N^r = 0$ Assume that i_r is the queried column, it holds that $\sum xi{=}1\, eidij = \sum ri{=}1\, Nldirj = (dirj)$N
Above all, all the elements in the queried i_rth column are obtained.

Soundness of TDICP

Then, we further prove the soundness of TDICP by introducing the following game.
Assume there is an adversary A that corrupts on average ρ_{adv} blocks of an outsourced file, and succeed in the soundness game of the proposed protocol with the probability of δ. In the following proof, we show that if the query times g exceeds a threshold γ_{neg}, our protocol can recover the whole file with a probability of more than 1 − $\_\_^{n}{}_{r}$, where τ is the security

2 parameter, when there exists an adversary A that can succeed in the soundness game with the probability $\delta \geq \delta_{neg}$ = $\_\__{r}$.

2

Remind that n is the length of 1 the notes and s is the number of the notes in the note set, we first quantify δ with respect to the parameter ρ_{adv}. In order to succeed in the soundness game, the adversary A can perform under the following two conditions. 1) it does not corrupt any note; 2) it corrupts some of the notes, but can still provide valid notes that conform to the hidden function. Therefore, we define the probability that the adversary A can succeed in the soundness game with respect to
$\rho adv\ as$: $\rho = P(ASuccess,i) = (1 - \rho adv) + \rho 2advns$.

In TDICP, the integrity check requires the adversary A□ to response γ□ valid note sets to succeed in the soundness game, therefore

$$\delta = \sum_{i=1}^{\gamma} P^{A}_{(Success,i)} = (1-\rho_{adv})^{\gamma} + \underbrace{\frac{\gamma\rho_{adv}(1-\rho_{adv})^{\gamma-1}}{2^{ns}} + o\left(\frac{1}{2^{ns}}\right)}_{\xi}$$

Note that if ns is large enough, i.e., ns = 128, will then be negligible. We can simplify the above equation that if ns ≥ 128, $\delta \approx (1 - \rho_{adv})^{\gamma}$.

We then define a threshold ρ_{neg} with respect to
ρ_{adv} that if $\rho_{adv} > \rho_{neg}$, the probability of our protocol that fails in recovering the blocks is negligible. Since TDICP adopts ECC and can recover
d
ρD = errors, then for each block, if there
2 exists more than corrupted ^{d} errors, our
2
protocol fails in recovering the blocks. Let
$P_{(^{\sigma}Fail,i)}$ be the probability that a block has more than ^{d} errors. According to Chernoff bounds,
2
we can bound $P_{(^{\sigma}Fail,i)}$ as

$$P^{\sigma}_{(Fail,i)} \leq exp\left(-\frac{\rho_{adv}D}{3}\left(1-\frac{\rho}{\rho_{adv}}\right)^2\right)$$

Next, we define a threshold γ_{neg} for the query time γ that if an adversary A corrupts more than ρ_{neg} query time γ that if an adversary A corrupts more than ρ_{neg} fraction of the blocks, it will be detected by our protocol with an overwhelming probability. In other words, if $\gamma > \gamma_{neg}$ and $\rho_{adv} > \rho_{neg}$, then the probability of the adversary A to succeed in the soundness game is negligible. Then

$$\delta = (1-\rho_{adv})^{\gamma} \leq (1-\rho_{adv})^{\gamma_{neg}} \leq \delta_{neg} = \frac{1}{2^{\tau}}$$

According to the equation in $x \leq x - 1$, when
$\rho adv > \rho neg$

$$\gamma_{neg} = \left\lceil \frac{\ln(2)\tau}{\rho_{neg}} \right\rceil \leq \frac{-\ln(2)\tau}{\ln(1-\rho_{neg})} \leq \frac{-\ln(2)\tau}{\ln(1-\rho_{adv})}$$

Finally, we define the probability of a file to be recovered. Since if there exists one block failing to be recovered, the whole file fails to be recovered. Let $\prod_{Fail}{}^{\epsilon}$ *be* the probability that the file fails to be recovered, then $\prod^{\epsilon}{}_{Fail} \leq \sum^{n}{}_{i=1} P_{(Fail,i)}$. Recovered is negligible, i.e., $P_{(^{\epsilon}{}_{Fail,i)}} \leq {}_{2}{}^{n}{}_{\tau}$. The probability of the files to be successfully recovered is

$$\prod_{Success}^{\epsilon} = 1 - \prod_{Fail}^{\epsilon} \geq 1 - \frac{n}{2^{\tau}}$$

Privacy of UDDCP

We further prove the privacy of UDDCP based on the irreversibility of the cuckoo filter. In UDDCP, the data holder is private, which leaks no information to the CSP about its private inputs. Since the data holder selects all values uniformly and at random, i.e., $\{r_1, . . . , r_{Nc}\} \leftarrow Z_n{}^*$, thus, $r_i{}^{inv}$ and $r_i{}^{'}$ are all random sequences. The data holder masks its inputs $A[i]$ to the CSP with random values $r_i{}^{'}$, so that CSP cannot obtain any other $H(y_i)$ of the data holder except for the intersection. The CSP is private which leaks no information to the data holder since we introduce a cuckoo filter to store the computation results a_i in filter generation phase. Due to the irreversibility of the filter, the data holder cannot obtain any other $H(x_i)$ except for the intersection.

Soundness of UDDCP

We prove the soundness of UDDCP by illustrating how it can solve all potential cheats the CSP can perform, including

- the CSP may provide unauthorized tags that are not from previous data holders or delete some stored tags driven by some profits;
- the CSP may provide wrong computation results of a_j or $C[i]$ to the AA or the data holder.

In UDDCP, the first cheat can be tested, since we employ AA to verify all the signatures and record the number of the CSP’s tag set. Unauthorized tags created by the CSP are easily found out and the CSP is audited to provide all the tags from previous data owners. The second cheat can also be tested, since we let AA to verify whether $\prod H(x_j) = (\prod a_j)^e$ holds, which can be proved correct according to the multiplication homomorphism of RSA. Wrong computations of any a_j or $C[i]$ can be detected by the AA.

Integrity Check

With its integrated support for configurable tag creation for integrity checks over encrypted data deduplication, it can ensure that duplication checks are accurate. In particular, we suggest an innovative Tag-adaptable Deduplication-upheld Uprightness Check Convention based on Confidential Data Recovery. This convention includes the presentation of another check tag, note set, which empowers various clients to make their own confirmation labels while as yet supporting label deduplication at the CSP. In any case, these assignments need changing a similar document into the indistinguishable check tag. While deduplication is impacted, uprightness check security is improved by forestalling savage power assaults when several users with access to the same cloudstored file provide distinct tags indicating their need for data integrity check.

Ensuring Originality

The preceding literature also fails to address a key security concern: the CSP's accuracy guarantee of data duplication check. A number of approaches encourage CSP deduplication without considering the possibility that the CSP may deceive users with a false duplication check result. The rationale for this is straightforward: using deduplication to conserve space allows the CSP to earn more money by charging consumers the regular storage cost instead of providing a welldeserved discount. Table 1 shows four examples of when the CSP handles a file storage duplication check.

Access to Confidential Data

Each "note" in the TDICP's note set is a randomly generated bit sequence that follows a function f. This new verification tag is being investigated. Using Private Information Retrieval, the note set is annexed to the documents. To guarantee information trustworthiness over the CSP with deduplication similarity, TDICP allows clients to add their own check labels. An original test and reaction strategy in view of Private Set Convergence is being examined by the UDDCP. This approach will permit the client, instead of the CSP, to decide if the record is copy first. This will keep the CSP from misleading the client about the duplication actually look at result while the file is being uploaded.

Deduplication of Data

VeriDedup is an improved version of our earlier deduplication technique that uses PSI and PIR to guarantee the accuracy of duplication checks and data integrity checks during encrypted data deduplication. To permit the information holder to decide if the transferred blocks are copies before the CSP, we incorporate a PSI-based challenge and reaction instrument into the duplication really take a look at technique, which is different from earlier work. Also, we utilize AA to make sure the CSP's calculations during the duplication check are correct, so it can't trick users into uploading previously saved data blocks.

IV. RESULT ANALYSIS

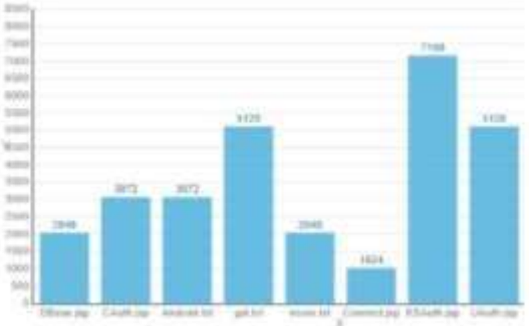


Figure 1: Data File Throughput

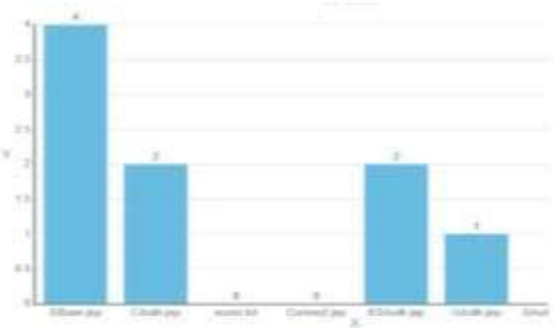
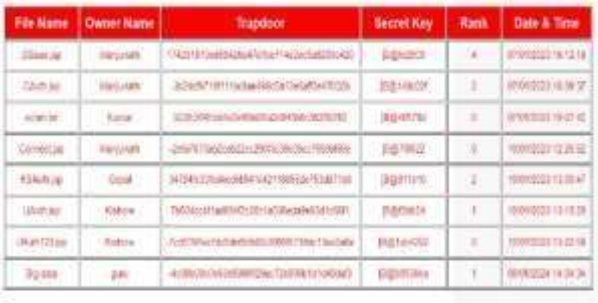


Figure 2: Data File Rank Results



File Name	Owner Name	Trapdoor	Secret Key	Rank	Date & Time
[illegible]	[illegible]	[illegible]	[illegible]	4	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	2	[illegible]
[illegible]	Kumar	[illegible]	[illegible]	0	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	0	[illegible]
[illegible]	Gopal	[illegible]	[illegible]	2	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	1	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	0	[illegible]
[illegible]	[illegible]	[illegible]	[illegible]	1	[illegible]

Figure 3: Uploaded Cloud Files

Id	Owner Name	File Name	MAC or HASH Code	Date & Time
1	[illegible]	[illegible]	[illegible]	[illegible]
2	[illegible]	[illegible]	[illegible]	[illegible]

Figure 4: Data Deduplication Files

V. CONCLUSION

With VeriDedup, you can ensure the accuracy of a duplicate check in an integrated manner and verify the integrity of an outsourced encrypted file. Multiple data holders may independently use their verification tags to validate the integrity of an outsourced file using VeriDedup's TDICP protocol, all without having to communicate with the data owner. However, to ensure the accuracy of the duplication check, we implemented a new challenge and answer method in VeriDedup's UDDCP duplication check protocol, which allows the data holder—rather than the CSP— to determine whether a file is duplicate. The results of the performance and security tests demonstrate that, when implemented using the specified security model, VeriDedup is both safe and effective. When compared to analogous earlier arts, our computer simulation results further demonstrate its efficiency.

Future Enhancement

- **Dynamic Data Updates**: Currently, VeriDedup might not support efficient updates to existing data. Developing a mechanism for users to securely update their data while maintaining integrity and deduplication benefits would be valuable.
- **Enhanced Privacy:** While VeriDedup protects data integrity, exploring techniques like homomorphic encryption could enable users to perform limited computations on their outsourced data without compromising confidentiality.
- **Decentralized Storage Integration:** Integrating VeriDedup with decentralized storage solutions could offer additional benefits like distributed trust and censorship resistance.
- However, ensuring compatibility and maintaining efficiency within decentralized architectures would be crucial.

REFERENCES

1. X. Chen, J. Li, J. Weng, J. Ma, and W. Lou, "Verifiable computation over large database with incremental updates," IEEE Trans. Computers, vol. 65, no. 10, pp. 3184– 3195, 2016.
2. M. Gerla, J.Wang, and G. Pau, "Pics-onwheels: Photo surveillance in the vehicular cloud," International Conference on Computing, Networking and Communications, pp. 1123–1127, 2013.
3. X. Chen, J. Li, J. Ma, Q. Tang, and W. Lou, "New algorithms for secure outsourcing of modular exponentiations," IEEE Trans. Parallel Distrib. Syst., vol. 25, no. 9, pp. 2386–2396, 2014.
4. H. Yuan, X. Chen, T. Jiang, X. Zhang, Z. Yan, and Y. Xiang, "Dedupdum: Secure and scalable data deduplication with dynamic user management," Inf. Sci., vol. 456, pp. 159–173, 2018.
5. H. Huang, X. Chen, Q. Wu, X. Huang, and J. Shen, "Bitcoinbased fair payments for outsourcing computations of fog devices," Future Generation Comp. Syst., vol. 78, pp. 850–858, 2018.
6. IDC. (2014) The digital universe of opportunities: Rich data and the increasing value of the internet of things. [Online]. Available: https://www.emc.com/leadership/digitalu niverse/2014iview/index.htm
7. W. J. Bolosky, S. Corbin, D. Goebel, and J. R. Douceur, "Single instance storage in windows 2000," in Conference on Usenix Windows Systems Symposium, 2000.
8. Dropbox. (2007). [Online]. Available: http://www.dropbox.com
9. GoogleDrive. (2012). [Online]. Available: http://drive.google.com
10. Memopal. (2018). [Online]. Available: http://www.memopal.com
11. Netapp. (2008) Netapp deduplication helps duke institute for genome sciences and policy reduce storage requirements for genomic information by 83 percent. [Online].Available: http://www.netapp.com
12. M. Dutch, "Understanding data deduplication ratios," in SNIA Data Management Forum, 2008, pp. 1–13.
13. T. Jiang, X. Chen, J. Li, D. S. Wong, J. Ma, and J. K. Liu, "TIMER: secure and reliable cloud storage against data re-outsourcing," Information Security Practice and Experience - 10th International Conference, pp. 346–358, 2014.