International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Secure and Privacy-Preserving Framework for Verifiable Data Deletion in Cloud Storage

B K Harish[1], E Pavithra[2]

[1]P.G Scholar, Department of MCA, Sri Venkatesa Perumal College of Engineering & Technology, Puttur,
E-mail: harishnaidu2113@gmail.com, ORC-ID: https://orcid.org/0009-0008-1113-748X
[2]Assistant Professor, Department of MCA, Sri Venkatesa Perumal College of Engineering & Technology, Puttur,
E-mail: epavithra526@gmail.com, ORC-ID: https://orcid.org/0009-0006-8871-4551

Abstract: Public cloud settings make it easier for many people to share data and work together, but they also make it harder to delete data in a way that can be tracked and is safe. Users can't be sure that their deletion requests will be carried out honestly because cloud service providers may keep deleted data in secret files that can be used by people who aren't supposed to. To fix this problem, the Verifiable Deletion Protocol (VDUP) is created. It improves data security by separating deletion requests from credential replies by using uncertainty roles and uncertainty requests. This system makes sure that pre- and post-deletion verification can't be told apart. This way, cloud services can't figure out who sent the request or connect it to credentials. The protocol formally defines security properties and uses specific examples and security proofs to show that it can withstand backup attacks. A three-step process is built into the system: Anonymity Check 1 is used for data fingerprinting and deletion request filing before the deletion, and Anonymity Check 2 is used for validation after the deletion. In addition, an add-on encrypts stored data with AES and sets up access control rules that let users choose whether a file is public or private. Experiments show that this method reduces the time and effort needed to create credentials compared to current ones. It also finds unauthorized storage by comparing signatures between check steps, which makes users more confident in the deletion of data in the cloud.

"Index Terms: *Cloud security, public cloud, verifiable behavior, verifiable deletion."*

1. INTRODUCTION

These days, cloud storage is an important part of modern computers because it lets a lot of different programs access data quickly and easily. It is important in many important areas, like healthcare, finance, government, and education, where huge amounts of private data are saved, retrieved, and sent all the time [1]. This change from managing data locally to managing data in the cloud has brought many benefits, including lower costs, more freedom, better teamwork, and uninterrupted business operations [2, 3]. But these benefits aren't as great as they could be because of ongoing worries about data security, especially the promise of full and permanent deletion of data when it's no longer required.

When people upload data to the public cloud, they usually give up direct control over it. Instead, they trust cloud service providers (CSPs) to manage, store, and finally delete the data safely [4]. This trust approach is flawed by nature, which is a shame. Because CSPs are only partially trusted, they might not follow deletion directions perfectly, either because of technical issues, company rules, or even bad intentions [5]. Even if data is marked to be deleted, it may still be kept for system backups, audit logs, machine learning analytics, or business reasons, and the user may not even be aware of it. Public scandals like the Facebook-Cambridge Analytica data scandal show how dangerous it is to keep data without permission and how there aren't any good ways to police the law [6].

The main problem is that there isn't enough openness and responsibility. At the moment, users can't be sure that deleted data has been completely and forever erased from all storage levels, such as replicas and distributed backups [7]. This is very bad for people's privacy and trust, and it's also bad for businesses that have to follow data protection rules like the General Data Protection Regulation (GDPR), which says people have the "right to be forgotten" and stresses how important it is to keep data as small as possible [8]. Not being able to confirm deletion makes it harder to follow the rules, increases legal risks, and could hurt the image of businesses that use cloud storage.

So, the idea of verified deletion is becoming more popular, with the goal of creating ways to prove that data has been permanently deleted through

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

cryptography or third-party auditability. Taking care of this worry is important for building trust in cloud infrastructure, giving users control over their digital assets, and making sure that the system works with global data governance standards.

2. LITERATURE REVIEW

Ozdemir et al. [6] showed a quick and easy way to use set accumulators for scaling verifiable computing. Their work is mostly about using cryptographic adders to help make proven computer systems that can be used in the real world. This makes them more scalable and useful. They pointed out that set accumulators can quickly check calculations on changing sets of data without needing all the data to be present at the time of checking. This is an important step toward making scalable checking systems work in the cloud.

One-way accumulators were first thought of by Benaloh and de Mare [7] as a decentralized option to digital signatures. Their groundbreaking work laid the groundwork for cryptographic structures that make it safe to check if data belongs to a set. This is very important for uses that need to be sure of integrity without having a single trust authority. This method works especially well when you want to make sure that a piece of data was removed or left out of a dataset correctly without showing the whole dataset.

Papamanthou et al. [8] suggested the best ways to check operations on dynamic sets. Their method makes sure that any change or query on a dataset can be checked with little extra work, which is good for both speed and soundness. This method works in changing places like cloud storage systems, where data is often changed, and the accuracy of these changes needs to be ensured by standards that can be checked.

In their paper [9], Campanelli et al. introduced incrementally aggregatable vector commitments (IAVCs) and showed how they can be used in autonomous storage. Their work is innovative because it combines small cryptographic promises with efficient update methods. This makes it possible to outsource storage in a way that is safe, can be checked, and requires little client-side computation. Their approach is very useful for making sure that users can trust the integrity of data stored remotely in cloud systems so that anyone can check it.

From lattice-based cryptography, Peikert et al. [10] made vector and functional promises. These promises are safe from quantum threats and make it easy to make proofs for complicated queries over datasets. Their model offers strong security promises and works with post-quantum cryptographic standards. This makes it compatible with the long-term needs for data protection in cloud infrastructure.

Wee and Wu [11] made the field even better by using lattice assumptions to come up with short promises for vectors, polynomials, and functions. Their work greatly increases the size of the proof and the time it takes to check it, which is good for low-latency apps. These additions are very important for storage and computation protocols that can be checked, especially when clients need to check server-side processes with as little delay and bandwidth as possible.

Zhang et al. [12] created clear polynomial delegation methods that are used in proof systems with no prior knowledge. Their design gets rid of the need for a trusted setting and makes things more open and easy to check. This feature of their plan makes it perfect for use in safe cloud storage and blockchain-based verification systems.

Chen et al. [13] suggested databases that can be checked by anyone and can handle all updating tasks efficiently. Any third party can check the database through their system without having to see the private data, which makes things more open. This feature solves the important problem of trusting outside data management, especially when it comes to following rules and saving records that can't be changed.

Wang et al. [14] created a tag-based delegated set intersection technique that can be checked and used with private datasets that were outsourced. Their model makes sure that intersecting datasets kept in different cloud domains can be checked safely without revealing private data. This makes collaborative analytics useful while protecting privacy.

Chen et al. [15] came up with proof-carrying data methods that are made up of random oracles that have been arithmetized. This new system allows for dynamic computation verification, so clients can check both the accuracy of the data and the security of computations that were outsourced. When service providers aren't honest, the plan protects against it very well.

Yang et al. [16] suggested a fast way to use authenticated matrix promises for verifiable databases

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

of any size. Their work lets people safely ask questions about very large numbers without having to pay a lot for computing or communication. This improvement is especially important for cloud storage systems that need to handle a lot of changing data and provide strong security and verifiability promises.

3. MATERIALS AND METHODS

The suggested system adds a safe and checkable deletion structure called Verifiable Deletion Protocol (VDUP) to make sure that data deletions in public cloud environments are correct. The main idea is to use uncertainty-based methods like uncertainty requests and uncertainty roles to separate deletion requests from user credentials. This way, the cloud won't be able to link a specific deletion action to a specific user. The protocol has three key steps: (1) Anonymous Check 1, where users collect file signatures before deleting them; (2) Deletion Request, where users delete certain files from the cloud; and (3) Anonymous Check 2, where users check file signatures again for errors. AES encryption is also used on shared files to keep data safe from people who shouldn't have access to it, even if there are security holes. Access control rules are also built into the system, which lets users mark files as either public or private. All of these improvements make deletion activities in cloud storage more private, secure, and easy to check.

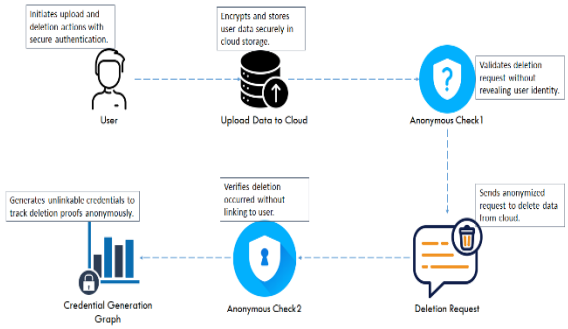


Fig.1 Proposed Architecture

The design of the system makes sure that deletions can be checked in the cloud while keeping users' identities secret. The user sends encrypted data to the cloud in a safe way. When someone requests to delete something, Anonymous Check1 checks the request without showing who it is. Then a Deletion Request is sent without saying who sent it. Anonymous Check2 makes sure that the user's data is deleted without connecting to them. To help with this, the Credential Generation Graph makes credentials that can't be linked to other credentials, so deleting proofs can be safely tracked. This approach makes sure that deletions in the cloud are private, authenticated, and can be checked.

a) Modules:

User Signup: New users or data owners can sign up by adding their credentials and personal information. This module securely saves user information in the database. After authentication, users can create accounts and use system features like uploading and deleting data and verifying their identities.

User Signin: Allows registered users to log in with their valid passwords. After proving their identity, the user is able to access the system's dashboard, where they can handle data files, do anonymous checks, and ask for deletions while still having safe user access.

Upload Data to Cloud: Gives users more security when they send files to the cloud. AES is used to secure the files, and verification hashes are linked to them. Access control can be used to keep other people from seeing certain files by marking them as public or private.

Anonymous Check1: Checks the file before it is deleted by getting information and signatures about it from the cloud secretly. These results are saved locally so that they can be used as a guide to check that the future delete request worked, which makes sure that the initial data capture was fair.

Deletion Request: It lets people pick out files in the cloud and delete them. It starts the deletion process on the cloud server without showing who made the request. This protects privacy [17] and gets the system ready for verification after deletion.

Anonymous Check2: After elimination, it does a second anonymous check. It checks the signatures of the files we have now against those from Anonymous Check1. If differences are found, deletion is verified; identical signatures suggest that the cloud may keep the backup.

Credential Generation Graph: Makes and shows a graph that compares the amount of extra work that needs to be done on computers for existing systems and the suggested protocol. It shows how well the VDUP protocol works to cut down on the time it takes to make credentials and boost system speed.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

Logout: Ends the current session safely, making sure that no one else can get in after the user leaves. It deletes all session data and sends users back to the login page, which keeps the system safe and private.

b) Methods/Technologies:

AES Encryption: The Advanced Encryption Standard (AES) is a symmetric encryption method that is used to keep files safe before they are sent to the cloud. It makes sure that even if cloud storage is hacked, people who aren't supposed to can't read the protected files. This method makes data more private and stops people from misusing or stealing private user data kept on cloud servers.

Hashing (File Verification Hashes): Cryptographic hash functions are used to make unique digital signatures for each shared file. During both private checks, these fixed-size hashes help make sure that files are correct and real. Any change in the content of a file creates a new hash. This lets users find changes that weren't made by them, files that were removed but were later recovered, or cloud backups that were made after a deletion request.

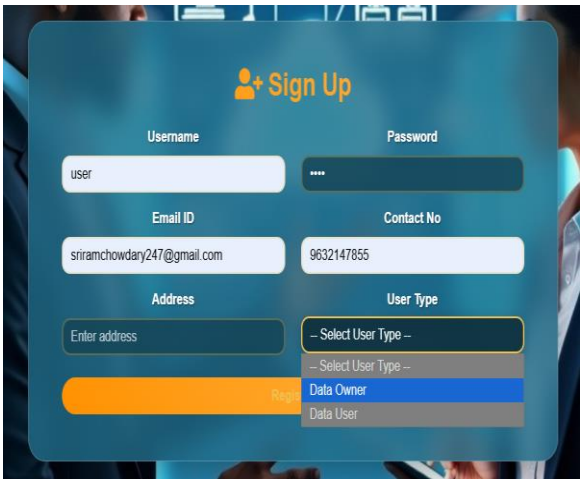
Anonymous Verification (Uncertainty Requests): Sending uncertainty-based data requests to the cloud for verification hides the name of the person making the requests. This is done before and after a deletion action to make sure that the verification is fair and can't be linked to anything else. It's impossible for the cloud to tell which request is related to deletion, which makes the checking process more private and trustworthy [19].

Credential Decoupling Technique: Credential decoupling separates user identity and credential information from deletion actions using uncertainty roles. With this method, the cloud can't connect requests to delete files to a specific person or file. It makes it harder for the cloud provider to target data retention and identity-based tracking, which protects data privacy and makes sure that deletions are real.

Comparison-Based Signature Validation: To use this method, you need to compare the file signatures that were obtained during Anonymous Check1 and Check2. If the hashes are different, the file was removed successfully. If they are the same, it means the cloud kept a copy. It lets users successfully confirm deletion and find bad behavior in the cloud by checking to see if the file still exists after a deletion request [20].

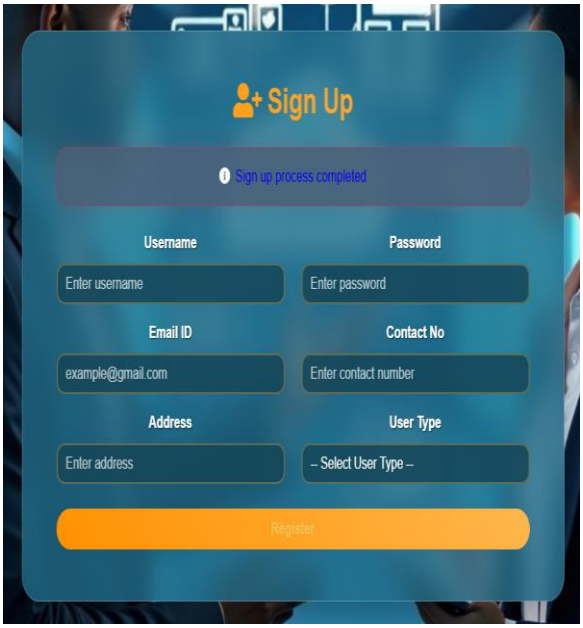
4. EXPERIMENTAL RESULTS

After that, the project window shows up like in the picture above.

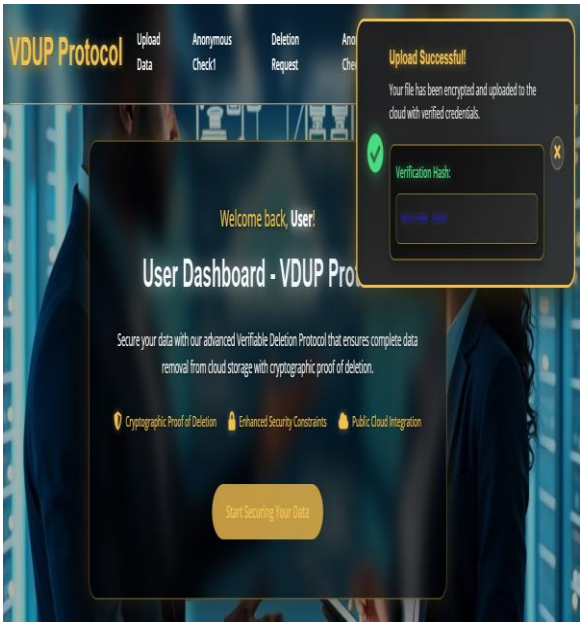


Complete all the needed fields to become a user, and then choose the type of user shown below.

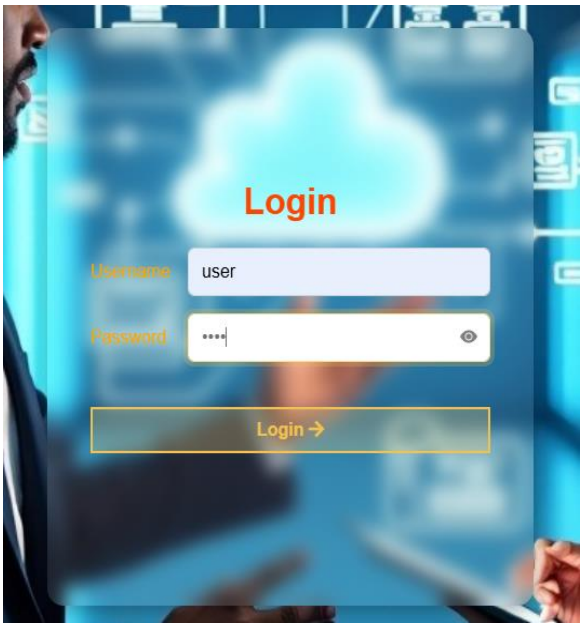
International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper



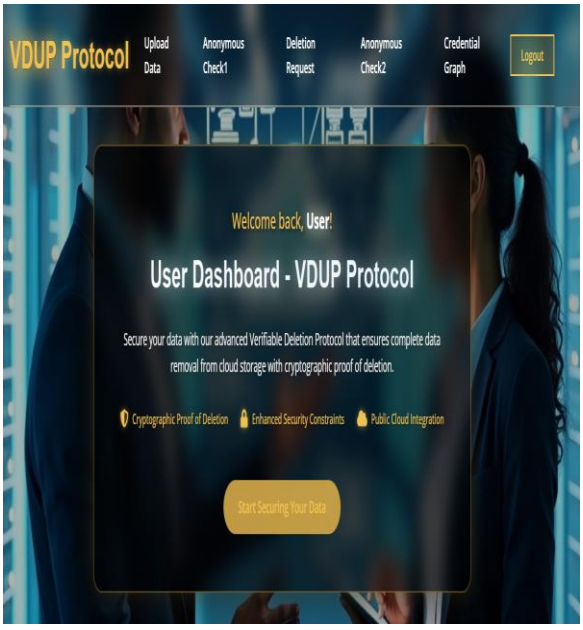
Registration of users has been finished successfully, and the data is now safely stored in the database.



Once a user logs in successfully, their information is encrypted and saved safely in the cloud.

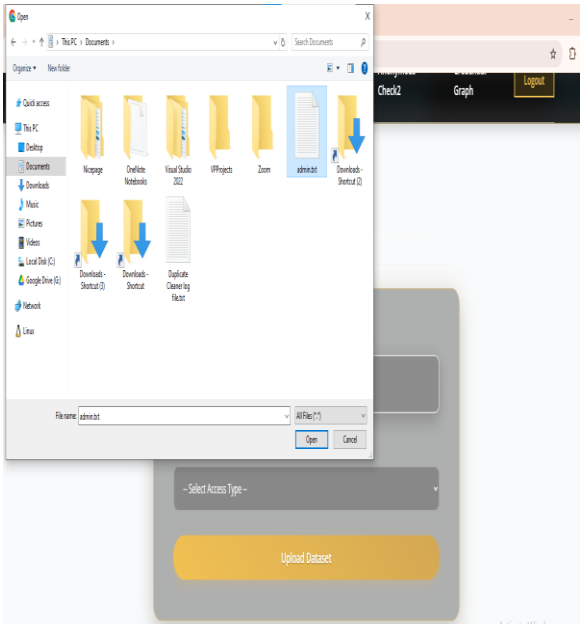


After you're done with registration, enter and send in the correct password information that appears on the screen.



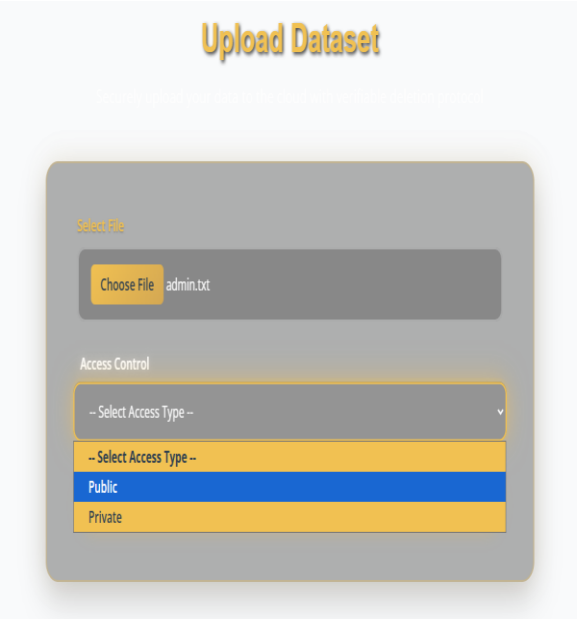
The person is taken to the home page after successfully logging in with their username and password.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

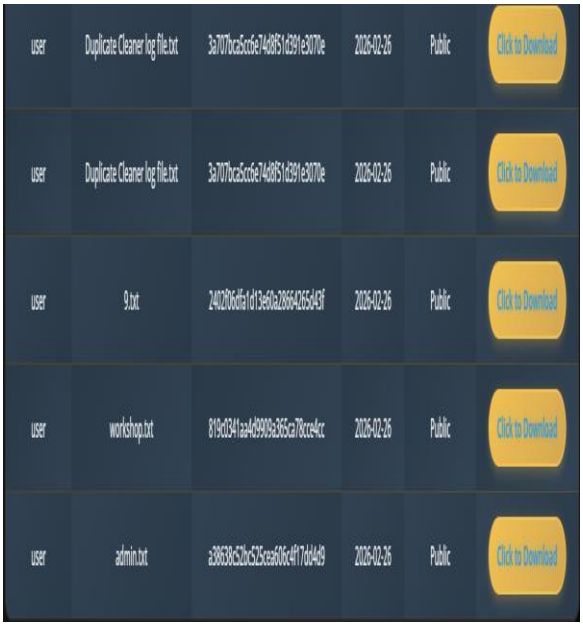


To meet different needs, you can pick a file from your local storage and send it to the cloud server.

The system makes the verification hash and shows it along with the file verification state in the cloud once the file is uploaded.

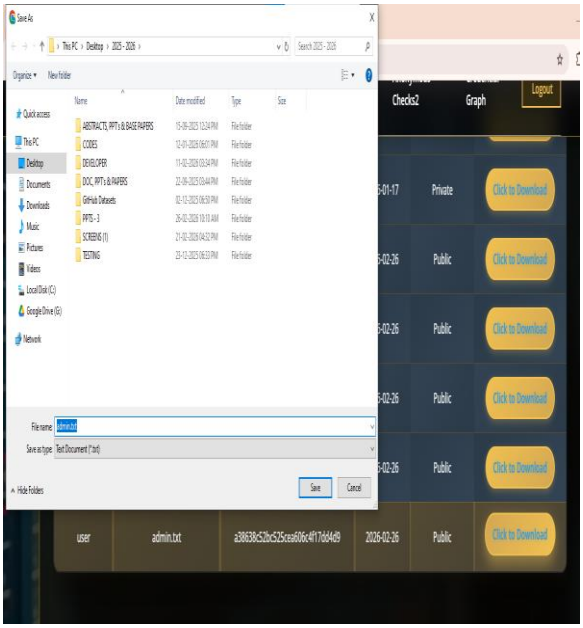


You can choose between public and private cloud access to manage who can see what when they need to.



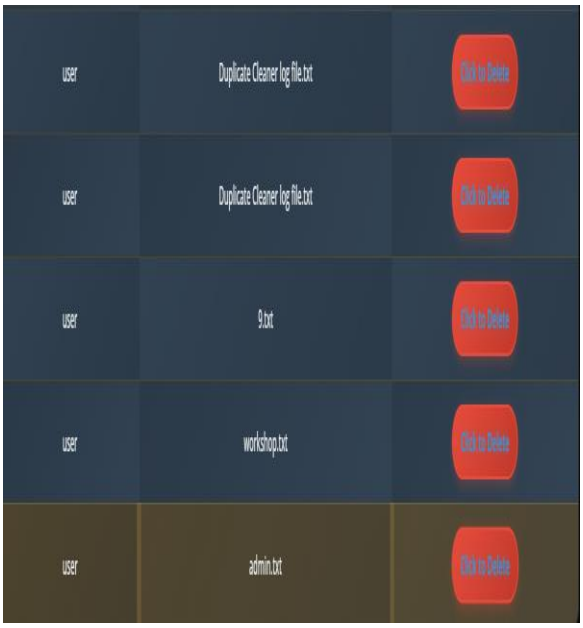
The system lets users see details about uploaded files, like the hash address, and download the files when they need to.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

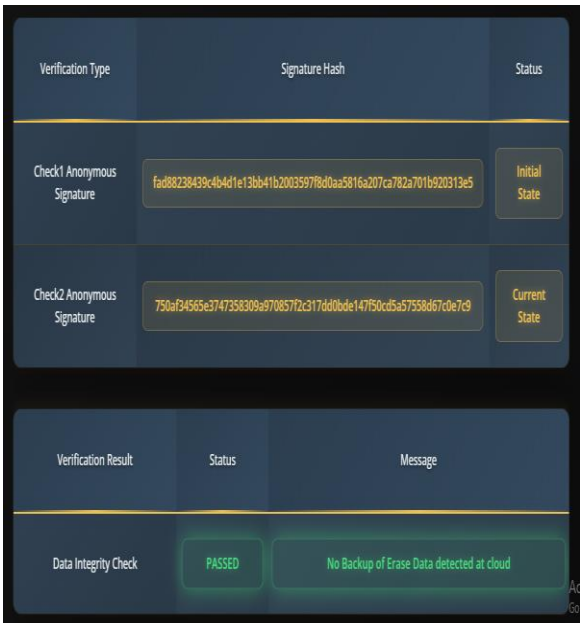


The file is moved from the cloud server to the local machine as soon as the download button is pressed.

When the "Delete" button is pressed, the file is removed from the cloud storage system for good.

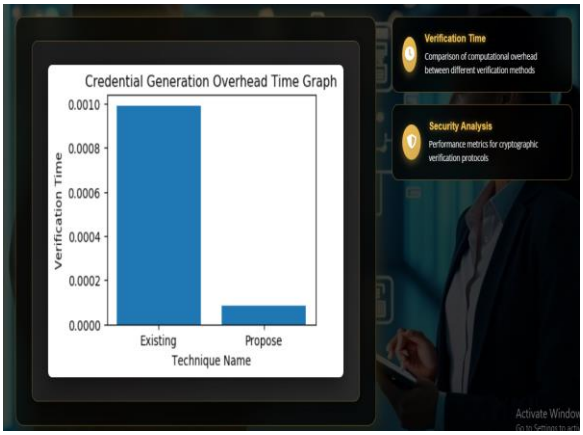


Users can see the files they've posted and delete them from cloud storage whenever they need to.



Check to see if the deleted file is still intact on the cloud service after you've deleted it.

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper



The x-axis shows the name of the technique, and the y-axis shows the proof time.

5. CONCLUSION

The Verifiable Deletion Protocol (VDUP) successfully makes sure that users can freely check that data deletion requests in public clouds are legitimate, without having to trust the cloud service provider. The system breaks the direct link between deletion commands and user credentials by adding uncertainty roles and requests. This makes it much harder for the cloud to fake or change deletion proof. This feature improves openness and responsibility by letting users find out when data is being kept without permission, even when the cloud tries to hide this by using backup copies.

Adding AES encryption adds an extra layer of security, making sure that data can't be accessed without the right decoding keys, even if it is stolen or backed up without permission. Access control methods also let data owners tell the difference between public and private files, limiting who can see and access them based on their user.

Overall, the work provides a strong solution that checks how deletions are handled, encrypts data to keep it safe, stops backups from being misused, and sets strict access limits. This creates a complete security framework for reliable cloud data management.

In the future, AI-driven anomaly detection could be added to the protocol to allow real-time alerts for unauthorized entry or backup attempts. Integration with blockchain can improve transparency even more by permanently recording deletion actions. Scalability can be improved by allowing multiple clouds to work together and platforms to talk to each other. Data will be safer if user interfaces are made easier to use and encryption methods are expanded to include hybrid or quantum-safe techniques. These changes will make the system stronger, more flexible, and able to be used in a wider range of cloud-based security situations.

REFERENCES

[1] Tian, T., Liu, Z., Gao, T., Zhang, X., Jin, S., & Liu, X. (2025, April). Blockchain-based verifiable data deletion and software management for cloud storage. In Fifth International Conference on Telecommunications, Optics, and Computer Science (TOCS 2024) (Vol. 13629, pp. 513-519). SPIE.

[2] Lapmoon, J., & Fugkeaw, S. (2025). A Verifiable and Secure Industrial IoT Data Deduplication Scheme With Real-Time Data Integrity Checking in Fog-Assisted Cloud Environments. IEEE Access.

[3 Ganesh, B. R. ., B M, P., Prasad K, K. ., Swapna, G., & G, Viswanath. (2025). Data Mining-Driven Multi-Feature Selection for Chronic Disease Forecasting. Journal of Neonatal Surgery, 14(5S), 108–124. https://doi.org/10.52783/jns.v14.1993

[4] Ali, M., & Liu, X. (2025). A Novel Approach to Cloud Security: Publicly Verifiable Remote Signcryption Framework. IEEE Internet of Things Journal.

[5] A D Venkatesh, K Bhaskar, G Swapna, & G Viswanath. (2025). Advanced Hybrid Learning Architecture for Precision Cardiovascular Risk Assessment. In International Journal of Health Sciences and Pharmacy (IJHSP) (Vol. 9, Number 1, pp. 50–61). Zenodo. https://doi.org/10.5281/zenodo.15448632

[6] Ozdemir, A., Wahby, R. S., Whitehat, B., & Boneh, D. (2020). Scaling verifiable computation using efficient set accumulators. *Proceedings of the 29th USENIX Security Symposium*, 1–12.

[7] Benaloh, J., & de Mare, M. (1994). One-way accumulators: A decentralized alternative to digital signatures. *Workshop on the Theory and Application of Cryptographic Techniques*, 274–285.

[8] Papamanthou, C., Tamassia, R., & Triandopoulos, N. (2011). Optimal verification of operations on dynamic sets. *Annual International Cryptology Conference*, 91–110.

[9] Campanelli, M., Fiore, D., Greco, N., Kolonelos, D., & Nizzardo, L. (2020). Incrementally aggregatable

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

vector commitments and applications to verifiable decentralized storage. *International Conference on the Theory and Application of Cryptology and Information Security*, 3–35.

[10] Peikert, C., Pepin, Z., & Sharp, C. (2021). Vector and functional commitments from lattices. *Proceedings of the 19th International Conference on Theory of Cryptography*, 480–511.

[11] Wee, H., & Wu, D. J. (2023). Succinct vector, polynomial, and functional commitments from lattices. *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 385–416.

[12 G Loge, T Sunil Kumar Reddy, G Swapna, & G Viswanath. (2025). Interpretable AI for Precision Brain Tumor Prognosis: A Transparent Machine Learning Approach. In International Journal of Health Sciences and Pharmacy (IJHSP) (Vol. 9, Number 1, pp. 180–195). Zenodo. https://doi.org/10.5281/zenodo.15523628

[13] Chen, X., et al. (2021). Publicly verifiable databases with all efficient updating operations. *IEEE Transactions on Knowledge and Data Engineering*, 33(12), 3729–3740.

[14] Wang, Q., Zhou, F., Xu, J., & Peng, S. (2022). Tag-based verifiable delegated set intersection over outsourced private datasets. *IEEE Transactions on Cloud Computing*, 10(2), 1201–1214.

[15] Chen, M., Chiesa, A., Gur, T., O'Connor, J., & Spooner, N. (2023). Proof-carrying data from arithmetized random oracles. *Proceedings of the 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques*, 379–404.

[16] Yang, H., Feng, D., & Qin, J. (2023). Efficient verifiable unbounded-size database from authenticated matrix commitment. *IEEE Transactions on Dependable and Secure Computing*, 20(5), 3873–3889.

[17] Xu, R., Li, C., & Joshi, J. (2023). Blockchain-based transparency framework for privacy preserving third-party services. *IEEE Transactions on Dependable and Secure Computing*, 20(3), 2302–2313.

[18] Liu, D., Huang, C., Ni, J., Lin, X., & Shen, X. S. (2022). Blockchain-cloud transparent data marketing: Consortium management and fairness. *IEEE Transactions on Computers*, 71(12), 3322–3335.

[19] G Ganesh, G Viswanath, G Swapna, & K Yatheendra. (2025). AI-Driven Hematological Analysis for Proactive Dengue Diagnosis. In International Journal of Health Sciences and Pharmacy (IJHSP) (Vol. 9, Number 1, pp. 196–210). Zenodo. https://doi.org/10.5281/zenodo.15541467

[20] Lu, J., Li, H., Liu, C., Li, L., & Cheng, K. (2022). Detecting missing-permission-check vulnerabilities in distributed cloud systems. *Proceedings of the ACM SIGSAC Conference on Computer and Communications Security*, 2145–2158. https://doi.org/10.1145/3548606.3560589