Veridedup: a verifiable cloud data deduplication scheme with integrity and duplication proof

[1]CH. ROHINI [2]N. KOTESEWARA RAO

[1]TM Department of CSE, Narayana Engineering College, Gudur

[2]Associate. Professor, Department of CSE, Narayana Engineering College, Gudur

ABSTRACT:

Data de-duplication 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. Data de-duplication is a technique to reduce the amount of storage space for each entity to save its data. It is a method of eliminating a dataset's redundant data. In a secure data de-duplication process, a de-duplication assessment tool identifies extra copies of data and deletes them, so a single instance can then be stored. 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. It is 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. Any data hosted by cloud providers is protected with encryption, allowing users to access shared cloud services conveniently and securely.

Keywords: Integrity check, duplication check, private information retrieval, data deduplication, cloud computing, verifiable computation

1. INTRODUCTION:

Internet users have increased dramatically in the past few years. More and more people are coming online. It has become an important part of their daily life and connected with their social lives. Greater number of jobs, businesses, educational institutions are taking benefit of technology to get the work done even in such hard times.

The changes it has brought are here to stay, but such advancements in technology also expose it to some serious security threats which makes cyber threats one of the biggest global risks. In the past few weeks of pandemic online threats have risen as much as six times.

Attackers try to steal or modify our data and can even take control of our systems. We have examples of global-scale attacks like ransom ware and some attacks on the zoom platform. Despite of various choices available

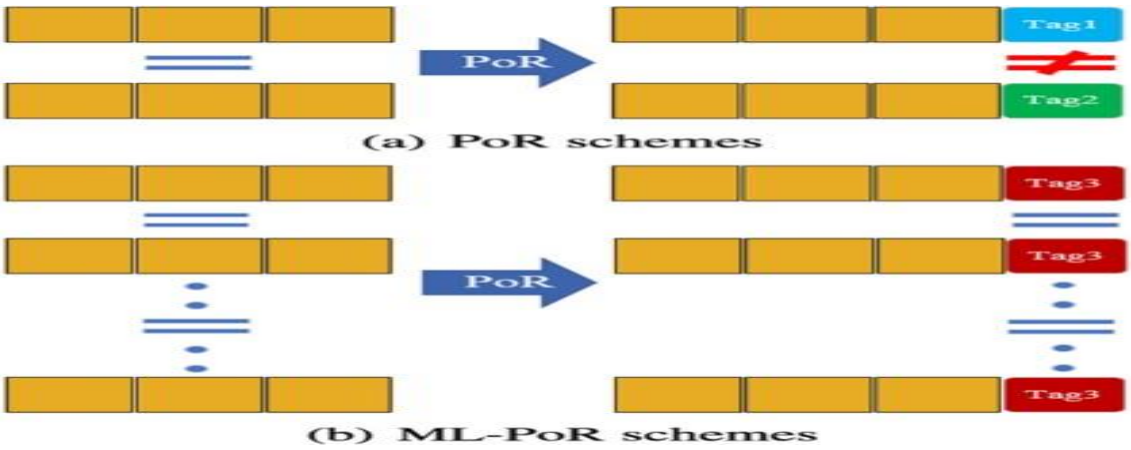
for data storage including cloud data storage, one of the major challenges faced by users & organizations is about data duplication.

It has been observed that for a single user or single transaction, there is a lot of duplication of data resulting due to usage of different sources of information. Data de-duplication is one of the effective techniques for data reduction. This technique ensures storage one single copy of each data. This is possible by comparison of data fingerprints with the existing stored data and thus identifies duplicate data.

Due to technological advancement, the volume of data owned by any organization and individual is constantly increasing and this has led to the growing demand for storing this ever-increasing amount of data. This demand of storing huge volume of data cannot be satisfied by conventional data storing methods which used physical storage such as flash drives and hard disks.

2. RELATED WORK

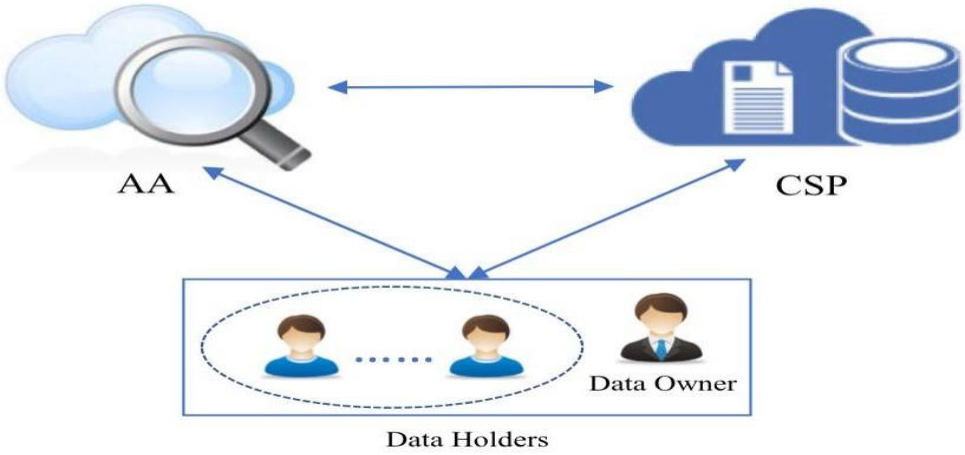
Our work is most related to the proof of retrievability (PoR) solutions in cloud data deduplication [15], [16], [17]. Juels and Kaliski [15] proposed a sentinel-based PoR scheme, in which a data owner adopts Error Correcting Code (ECC) and inserts special blocks called sentinels. The sentinels are indistinguishable from encrypted blocks in an encrypted file. During integrity challenge, a verifier asks a prover for those sentinels by disclosing their positions to the verifier. Therefore, this solution supports a limited number of PoR queries and after several times of queries, a data owner has to download the whole file and insert new sentinels to it. Ateniese et al. [6] proposed a scheme by defining the concept of Provable Data Possession (PDP) based on homomorphic tags, which is weaker than PoR in the way that it can verify that the CSP possesses parts of the file (called blocks) but cannot guarantee that the file is fully stored. However, the usage of homomorphic tags incurs high computation cost, which brings heavy computation burden to the data owner. Their later work [7] cooperates with an erasure code to help recover small corruptions. However, their solution suffers from such an attack that CSP can selectively delete some of redundant blocks but still can succeed in providing valid proof to the data owner



1 Proxy Re-Encryption (PRE): A PRE scheme consists of five polynomial time algorithms: Key generation(KG), Encryption(E), Re-encryption key generation(RG), Re-encryption(R) and Decryption(D): ðKG; E; DÞ are the standard key generation, encryption and decryption algorithms. Suppose we have two parties A and B. On input the security parameter 1k, KG outputs two public and private key pairs ðpkA; skAÞ and ðpkB; skBÞ. On input pkA and data M, E outputs a ciphertext CA ¼ EðpkA;MÞ. On input ðpkA; skA; pkBÞ, the re-encryption key generation algorithm RG outputs re-encryption key rkA!B for a proxy. On input rkA!B and ciphertext CA, the re-encryption function R outputs RðrkA!B; CAÞ ¼ EðpkB; MÞ ¼ CB. On input CB and skB, the decryption algorithm D outputs the plaintext M ¼ DðskB; CBÞ.

2 RSA-PSI PSI: [26], [27], [28] enables two parties to compute the intersection of their inputs in a privacy-preserving way, such that only their common inputs are revealed. A PSI scheme based on RSA blind signature (RSA-PSI) consists of four main phases: base phase, setup phase, online phase, and update phase: Base phase: Suppose we have a client C and a server S. S and C agree on the RSA public key ðN; eÞ and the false positive rate for the cuckoo filter CF[29]. S generates the RSA private key d, C chooses Nmax c random numbers and calculates their inverses as well as their modular exponentiation to the power e. Setup phase: On input the set owned by S, S encrypts it using its private key d and inserts the ciphertexts into the cuckoo filter and sends the CF to C. Online phase: C first blinds its inputs with the encryption of the respective random values and sends the resulting values to S. S responds the result to C by encrypting the resulting values using its private key d. Using the inverse of the respective random values, C can then unblind the encrypted blinded values through multiplications by applying the property of RSA that xed

x mod N. C finally obtains the intersection by checking whether the unblinded encrypted elements are in the CF that was sent by S in the setup phase. Update phase: On inputs a new element ui to its input, S encrypts it using its private key d and decides an efficient option to insert it into CF and sends the updated CF to C.



3. IMPLEMENTATION:

MODULES:

- Cloud
- Admin
- User

MODULES DESCRIPTION:

➢ **CLOUD**

• In this module the cloud manager have the login verification after that they can view all the database like user details and storage details.

➢ **ADMIN**

- Admin also have the login verification process.
- Admin will give the access for the user to login and also the secret key for the file to the user will be generated by the admin

➢ **USER**

- User have the registration process after that it goes to the admin part for access.
- After updating activation from the admin the user will move to the next part of process.
- Then the user will select the file, if the file already exist it shows the alert of data duplication or it will send key request to the admin.
- After getting the key from the admin the user can upload and download the file using that key

4. SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

- Processor : Intel Core i5
- Hard Disk : 200 GB
- Monitor : 18’ LED color
- Mouse : DELL.
- Keyboard : 110 keys enhanced
- RAM : 3GB

Software Requirements

- Operating System : Windows 7 / 8 / 10

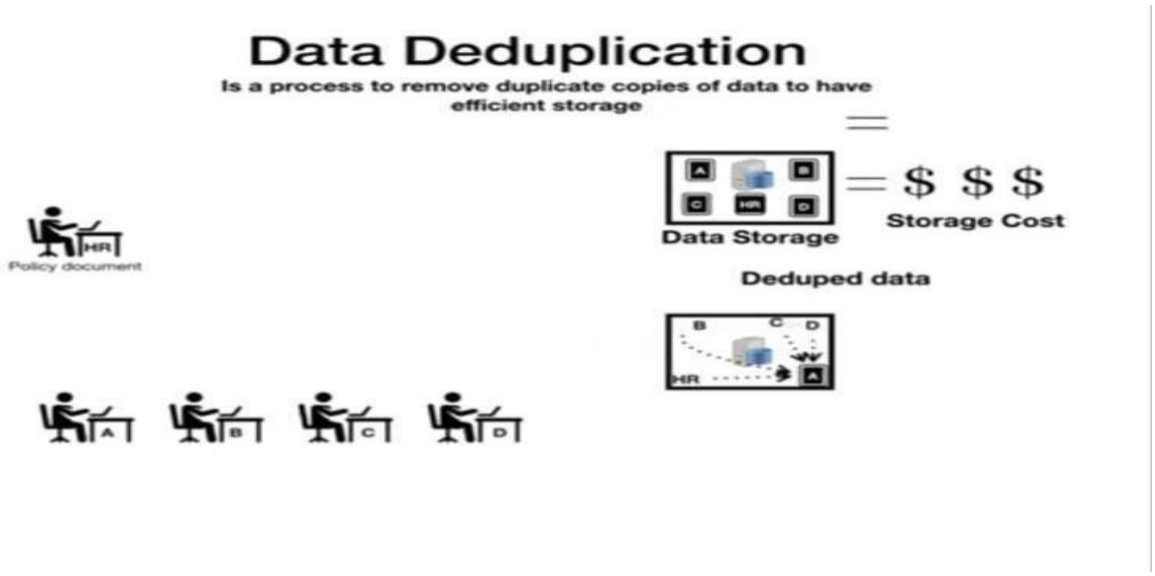
- Language Used : Java
- Database : My SQL
- User Interface Design : JFrame
- Server : Xamppserver

---

5. Methodology:

Problem Statement:

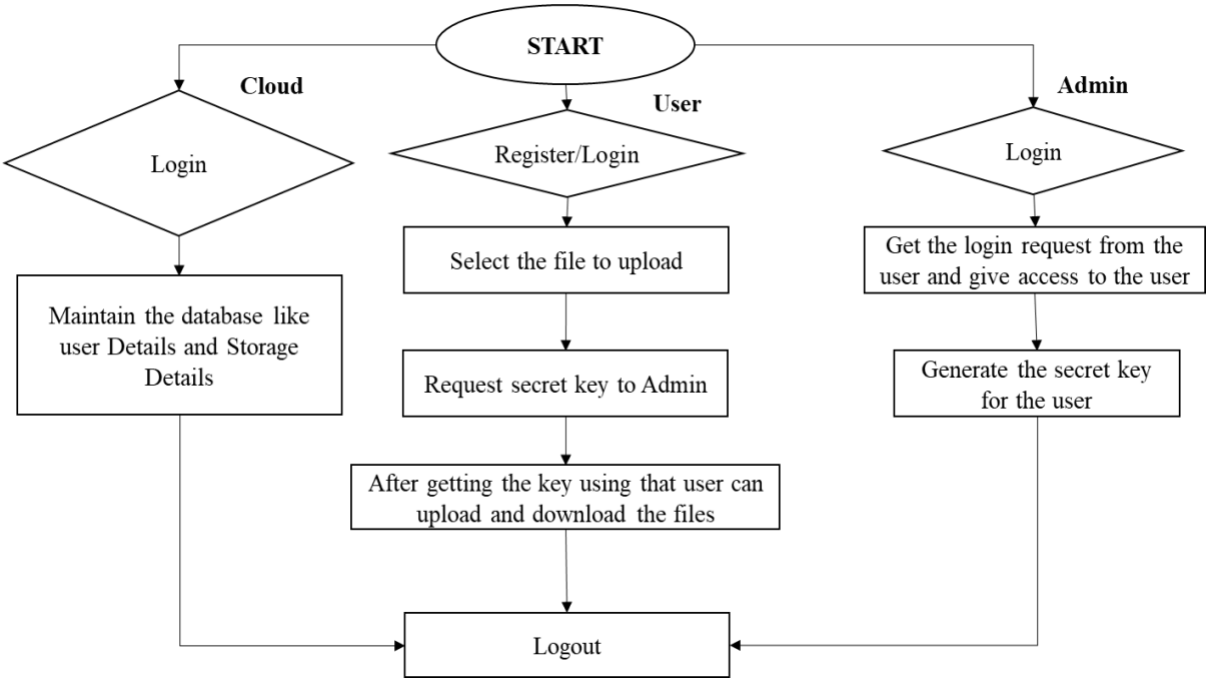
Data duplication is a common problem in data architecture that can affect the accuracy, consistency, and performance of data systems. Data duplication occurs when the same or similar data is stored in multiple locations or formats, creating redundancy, confusion, and potential conflicts. In our project we have



one many validations like checking the user details and storage details of the user, we can avoid the problem of duplicate data.

Data explosion has brought challenges to cloud storage management. To improve cloud storage efficiency and save network communication bandwidth, cloud data de-duplication has emerged as a research hotspot, especially in the field of encrypted cloud data storage. How to enhance the security of encrypted data de-duplication by resisting various attacks on de-duplication has become an important research issue. However, existing solutions suffer from security flaws and are vulnerable to a series of attacks, e.g., duplicate faking attacks, file ownership spoofing attacks, and file tampering attacks. Besides, dynamic data operation is rarely considered or

audited. To solve the above problems, we propose a novel scheme, named SecDedup, to enhance the security of encrypted cloud data de-duplication with dynamic auditing.

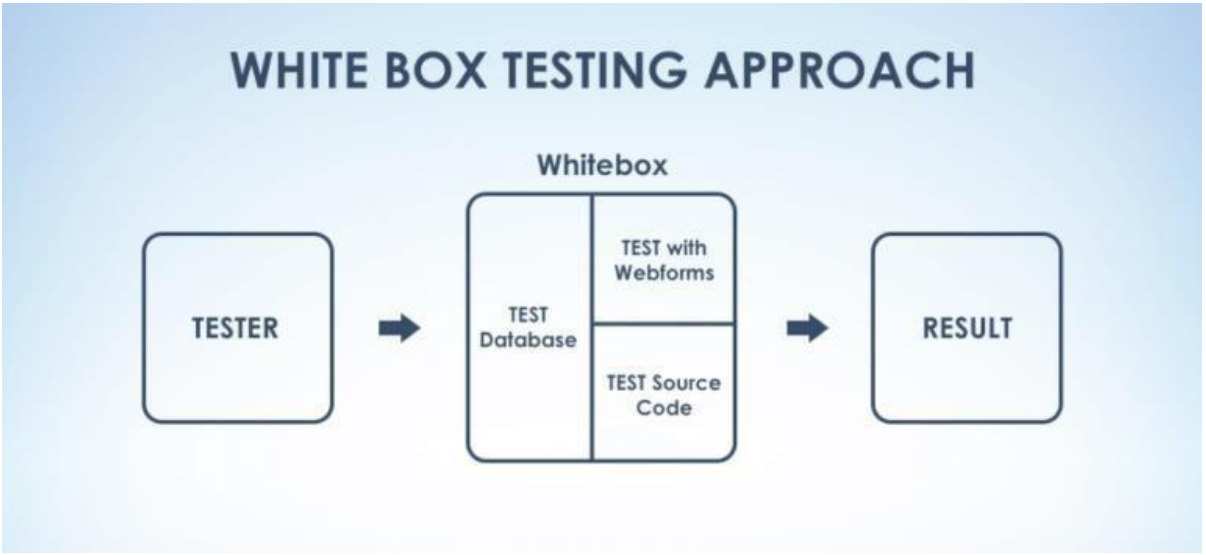


6. TESTING AND ANALYSIS:

- **WHITEBOX TESTING:**

White Box testing is a test case design method that uses the control structure of the procedural design to drive cases. Using the white box testing methods, we

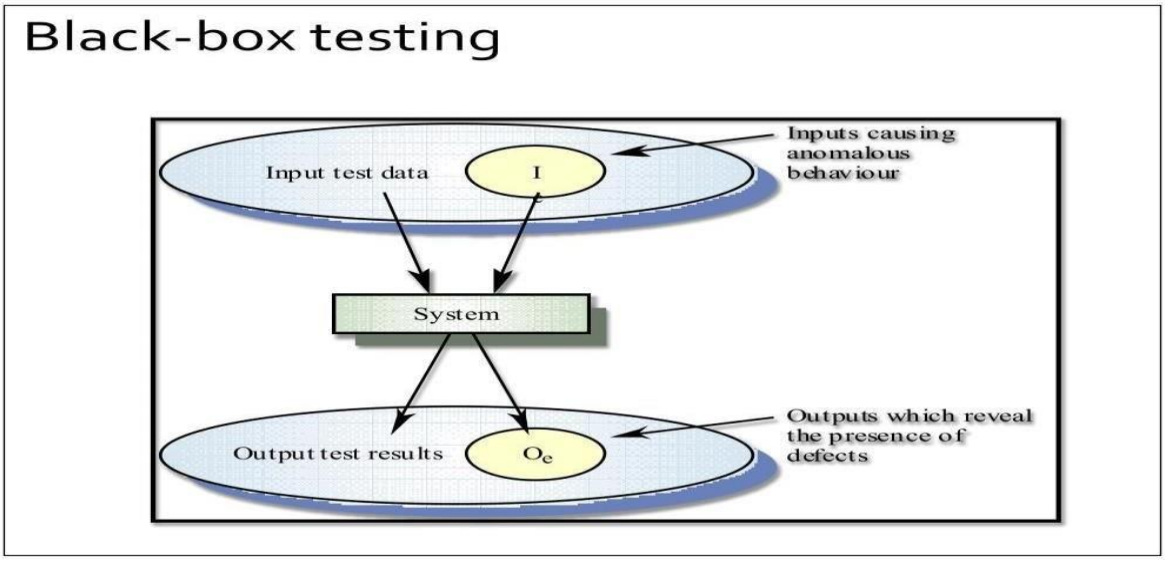
Derived test cases that guarantee that all independent paths within a module have been exercised at least once.



- **BLACK BOX TESTING:**

1. Black box testing is done to find incorrect or missing function
2. Interface error
3. Errors in external database access
4. Performance errors.
5. Initialization and termination errors

In 'functional testing', is performed to validate an application conforms to its specifications of correctly performs all its required functions. So this testing is also called 'black box testing'. It tests the external behaviour of the system. Here the engineered product can be tested knowing the specified function that a product has been designed to perform, tests can be conducted to demonstrate that each function is fully operational.



7. OUTPUT TESTING:

After performing the validation testing, the next step is output asking the user about the format required testing of the proposed system, since no system could be useful if it does not produce the required output in the specific format. The output displayed or generated by the system under consideration. Here the output format is considered

in two ways. One is screen and the other is printed format. The output format on the screen is found to be correct as the format was designed in the system phase according to the user needs. For the hard copy also output comes out as the specified requirements by the user. Hence the output testing does not result in any connection in the system.

8. CONCLUSION:

Data de-duplication is more than a space saving feature. It yields cost savings and efficiencies, in the long run, not to mention thwarting fraud risks significantly through identity verification services. It also cuts down compliance and regulatory risks. Though, the lack of standards for implementation and technology varies, making it tough to compare vendors.

REFERENCES:

[1] Z. Yan, L. Zhang, W. Ding, and Q. Zheng, "Heterogeneous datastorage management with de-duplication in cloud computing,"IEEE Trans. Big Data, vol. 5, no. 3, pp. 393–407, Sep. 2019.

[2] Z. Yan, W. X. Ding, and H. Q. Zhu, "A scheme to manageencrypted data storage with de-duplication in cloud," in Proc. Int.Conf. Algorithms Archit.

Parallel Process., 2015, pp. 547–561.

[3] Z. Yan, M. Wang, Y. Li, and A. V. Vasilakos, "Encrypted data management with de-duplication in cloud computing," IEEE Cloud Comput., vol. 3, no. 2, pp.

28–35, Apr. 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 Proc. 2nd ACM Conf. Data Appl. Secur. Privacy, 2012, pp.

[6] A. Giuseppe, R. Burns, and C. Reza, "Provable data possession at untrusted stores," in Proc. 14th ACM Conf. Comput. Commun. Secur., 2007, pp. 598–609.

[7] G. Ateniese et al., "Remote data checking using provable data possession," ACM Trans. Inf. Syst. Secur., 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 Proc. Int. Conf. Intell. Netw. Collaborative Syst., 2014, pp. 85–90.

[9] P. Meye, P. Raı¨pin, F. Tronel, and E. Anceaume, "A secure two phase data de-duplication scheme," in Proc. IEEE Int. Conf. High Perform. Comput. Commun., IEEE 6th Int. Symp. Cyberspace Saf. Secur., IEEE 11th Int. Conf. Embedded Softw. Syst., 2014, pp. 802–809.

[10] D. Vasilopoulos, M. Onen, K. Elkhiyaoui, and R. Molva, "Message- € locked proofs of retrievability with secure de-duplication," in Proc. ACM Cloud Comput. Secur. Workshop, 2016, pp. 73–83.

[11] M. Bellare, R. Canetti, and H. Krawczyk, "Keying hash functions for message authentication," in Proc. 16th Annu. Int. Cryptol. Conf. Adv. Cryptol., 1996, pp. 1–15.

[12] X. Liang, Z. Yan, X. Chen, L. T. Yang, W. Lou, and Y. T. Hou, "Game theoretical analysis on encrypted cloud data de-duplication," IEEE Trans. Ind. Informat., vol. 15, no. 10, pp. 5778–5789, Oct. 2019.

[13] X. Liang, Z. Yan, R. H. Deng, and Q. Zheng, "Investigating the adoption of hybrid encrypted cloud data de-duplication with game theory," IEEE Trans. Parallel Distrib. Syst., vol. 32, no. 3, pp. 587–600, Mar. 2021.

[14] Z. Yan, W. Ding, X. Yu, H. Zhu, and R. H. Deng, "De-duplication on encrypted big data in cloud," IEEE Trans. Big Data, vol. 2, no. 2, pp. 138–150, Jun. 2016.

[15] A. Juels and B. S. Kaliski, "Pors: Proofs of retrievability for large files," in Proc. 14th ACM Conf. Comput. Commun. Secur., 2007, pp. 584–597.

[16] J. Xu and E.-C. Chang, "Towards efficient proofs of retrievability,"in Proc. 7th ACM Symp. Inf. Comput. Commun. Secur., 2012, pp. 79–80.

[17] C. M. Tang and X. J. Zhang, "A new publicly verifiable data possession on remote storage," J. Supercomputing, vol. 75, no. 1, pp. 77–91,2019.

[18] H. Shacham and B. Waters, "Compact proofs of retrievability," in Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur., 2008, pp. 90–107.

[19] B. Dan, B. Lynn, and H. Shacham, "Short signatures from the weil pairing," in Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur., 2001,pp. 514–532.

[20] M. Azraoui, K. Elkhiyaoui, R. Molva, and M. Onen,"Stealthguard: €

Proofs of retrievability with hidden watchdogs," in Proc. Eur. Symp. Res.

Comput. Secur., 2014, pp. 39–256.