



ISSN: 2321-2152

IJMECE

*International Journal of modern
electronics and communication engineering*

E-Mail

editor.ijmece@gmail.com

editor@ijmece.com

www.ijmece.com

SECURE CLOUD DATA DEDUPLICATION WITH EFFICIENT RE-ENCRYPTION

¹SMT.S ARUNA, ²MYLA SURESH KUMAR

¹(Assistant Professor), MCA, Swarnandhra College

²³MCA, scholar, Swarnandhra College

ABSTRACT

Automated self-driving automobiles, etc., are a result of computer vision advancements that aid drivers. About 20% of accidents are caused by sleepiness or exhaustion on the part of the driver. It presents a significant challenge that has prompted several solutions to be suggested. Having said that, they can't handle processing in real time. These approaches suffer from a lack of robustness when it comes to dealing with lighting circumstances and variations in human faces. We want to install a smart processing system that may significantly lessen the occurrence of traffic accidents. Using this method, we may detect the driver's facial features, such as the frequency of blinking, the eye-mouth aspect ratio, the proportion of closed eyes, the frequency of yawning, the movement of the head, and so on. A camera is used in this system to continually observe

the driver. Haar cascade classifiers are used to identify the driver's eye and face. In order to determine whether both eyes are closed, we extract pictures of the eyes and feed them into a custom-designed convolutional neural network. The eye closure score is computed using Commercial cloud storage providers' extensive use of data deduplication techniques is crucial for keeping up with the exponential development of data. Various secure information deduplication methodologies have been created and utilized in various settings to support the wellbeing of clients' delicate information while put away in a re-appropriated way. Specifically, a few scientists have zeroed in on protected and quick re-encryption for scrambled information deduplication, and various frameworks have been created to work with dynamic proprietorship the board. This study exhibits that the recently evolved

lightweight rekeying-mindful encoded deduplication plot (REED) is powerless to an assault known as the stub-saved assault, with an accentuation on the re-encryption deduplication capacity framework. Moreover, utilizing the joined win big or bust change (CAONT) and haphazardly chose bits from the Sprout channel, we give a protected information deduplication technique that incorporates effective re-encryption. Our strategy guarantees the security of information proprietors' delicate information and can endure the stub-saved assault in view of the inborn element of one-way hash capabilities. To sweeten the deal even further, information proprietors might lessen the calculation cost of the framework by basically re-encoding a little part of the bundle by means of the CAONT, as opposed to the total bundle. Our technique is both safe and efficient in re-encryption, according to experimental findings and security analyses. ng the categorization. There will be an alert that goes off if the driver is determined to be too sleepy.

1.INTRODUCTION

More and more people and businesses are opting to pay as they go to have their sensitive data stored by distant

cloud service providers, thanks to the proliferation of cloud storage. In 2020, the amount of data in the universe is projected to reach 44 zettabytes (ZB), or 44 trillion gigabytes (GB), according to a study sponsored by Dell EMC and conducted by the Internet Data Center (IDC). This equates to more than 5,200 GB for every individual. But cloud service providers are under a lot of strain from the ever-increasing data volume. One easy way to deal with it is to mandate that cloud service providers consistently increase their storage capacity to fulfill customers' demands for high-quality storage services.

However, there is a lot of computational and managerial overhead involved in the complete life cycle of the data that cloud service providers keep, especially material that is abundant and repetitive (like genetic data, music, and movies). The initial approach to this issue that Bolosky et al. suggested was data de duplication, which reduces redundant storage space and bandwidth by removing multiple copies and retaining just one. These days, many cloud storage services use data deduplication algorithms. Some examples include Dropbox, Google Drive, and Memopal.

According to research, the data de duplication approach may significantly decrease the amount of genomic data ($\geq 83\%$) and disk data ($\geq 90\%$) used by business applications. The data de duplication approach has several benefits, but it still has certain security issues that must be resolved. One common misconception is that cloud service providers are not completely trustworthy and may attempt to deduce and analyze data that has been outsourced to them. Most people who utilize the cloud encrypt their data before sending it to the provider in order to keep it private. Unfortunately, data de duplication becomes impossible when many users encrypt the same data using their private keys. This causes the same data to produce various cipher texts. The first workable method for achieving de duplication on cipher texts while protecting data secrecy was suggested by Douceur et al. Nevertheless, private information stored in the cloud is encrypted using a convergent key that remains unchanging and is generated from the information's hash esteem. It will direct the deactivated cloud client to utilize the held focalized key to get to the delicate information.

With regards to cloud security, client renouncement is a significant issue. To exhibit this thought, we will utilize the example of genomic research. Due of the monstrous measure of hereditary datasets, specialists frequently store them on the cloud. Platforms tailored to the management and analysis of genetic data have been launched by Google Genomics and Amazon. Yet, there is a need to safeguard some of the sensitive genomic data generated by illness sequencing initiatives. For instance, researchers will no longer have access to the genome datasets if they are no longer part of the genome project. Methods like group key distribution and re-encryption have tackled this issue. Those frameworks might deal with client renunciation and joining by re-encoding delicate information utilizing symmetric encryption (like AES-128 or AES-256) and afterward dispersing bunch keys to bunch clients. Excessively computational above will be squandered by the re-encryption approach, regardless of whether it encodes the entire message with a new key to protect the information's security. Cryptographic requirement of access limitations is probably going to cause restrictive costs in situations including even

a minor degree of strategy dynamism, as shown by William et al.

An encrypted deduplication storage system that is aware of rekeying and allows lightweight re-encryption was recently suggested by Li et al. (REED). By minimizing the computing cost of the system, information proprietors are simply had to re-scramble a little part of the bundle through the CAONT, instead of the total bundle. By and by, it is quite important that the REED is defenseless to the stub-held assault. This assault will be examined in Segment 3.2. To sum up, it works like this: a cloud client whose entrance has been renounced holds the last bytes of a bundle as a stub bundle. Then, at that point, utilizing the held stub bundle and the managed bundle (downloaded from the cloud specialist co-ops), the repudiated client can recuperate the plaintext. Therefore, successful re-encryption and safe powerful proprietorship the board of cloud clients are not adequately supported by the current suggested systems.

The Value We Provide. The aforementioned issues with efficient and safe re-encryption for de duplication storage

are further investigated in this work. We have three main contributions:

The improved encryption of the REED system has a security flaw, as we point out [25], [26]. In other words, the so-called stub-reserved attack is something that this article proposes, and it can easily exploit this approach.

Secure data de duplication with efficient re-encryption is our proposed approach, and a site selection mechanism based on Bloom filters is also available. The data owner may be certain that the revoked cloud user will not be able to access their sensitive data thanks to the new location selection approach and symmetric encryption. Data privacy is therefore guaranteed. Additionally, data owners are only needed to re-encrypt a tiny portion of the package using the CAONT, rather than the full package, which significantly reduces the computation cost of the method.

- We analyse the scheme's security and performance and find that it's both efficient and secure.

2.LITERATURE SURVEY

Rekeying for Encrypted Deduplication Storage:

A rekeying-mindful scrambled deduplication capacity framework, REED, is proposed in this paper. To empower protected and lightweight rekeying while at the same time keeping deduplication capacities, REED utilizes a deterministic variation of the go big or go home change (AONT). Dynamic access control is an extension of the system that provides two encryption algorithms that strike a balance between performance and security. One drawback is that rekeying cannot be used to remove user access.

Efficient Reencryption Techniques in Cloud Storage:

Various approaches have been explored in literature regarding efficient reencryption in cloud storage environments. Techniques such as proxy reencryption, key-homomorphic encryption, and attribute-based encryption offer promising solutions to update ciphertexts efficiently while maintaining data security. Research in this area emphasizes the importance of minimizing computational overhead and latency during reencryption processes.

Security Challenges in Cloud Data Deduplication:

Studies highlight the security challenges inherent in cloud data deduplication, particularly concerning privacy and confidentiality. Deduplication introduces the risk of data exposure due to shared storage and metadata leakage. Encryption-based solutions are proposed to mitigate these risks, but concerns remain regarding the impact on deduplication efficiency and storage overhead.

Dynamic Access Control Mechanisms in Cloud Storage:

Ensuring the security and integrity of data in cloud storage systems is greatly assisted by dynamic access control techniques. Two of the most researched methods are policy-based access control (PBAC) and attribute-based access control (ABAC). Safeguarding sensitive information and promoting productive teamwork, these systems provide for granular access control based on qualities or previously established criteria..

Integration of Java Servlets for Cloud Security:

Literature explores the integration of Java Servlets with cloud security measures to enhance the protection of data and services. Servlets can be leveraged to implement

secure communication protocols, access control mechanisms, and encryption/decryption functionalities. Research in this area focuses on developing robust and scalable security solutions for Java-based cloud applications.

Evaluation of Existing Reencryption

Techniques:

Comparative evaluations of existing reencryption techniques provide insights into their performance, scalability, and security characteristics. Studies assess factors such as computational overhead, communication latency, and resistance against attacks. Benchmarking experiments and simulations help identify the most suitable reencryption method for specific cloud storage scenarios.

Future Directions and Emerging Trends:

Future research directions include exploring advanced cryptographic primitives for efficient reencryption, such as homomorphic encryption and lattice-based cryptography. Additionally, integrating blockchain technology for decentralized and auditable reencryption processes shows promise in enhancing the security and transparency of cloud data management. Furthermore, the

adoption of machine learning techniques for proactive threat detection and response in cloud environments is an emerging trend worth investigating.

This literature survey provides a comprehensive overview of existing research and developments in secure cloud data deduplication with efficient reencryption. By synthesizing insights from diverse sources, it lays the groundwork for the design and implementation of your Java Servlet-based project, ensuring it addresses key challenges and leverages state-of-the-art techniques in cloud security and data management.

PRELIMINARY INVESTIGATION

The primary goal of any project's development should be to follow the project guide. There are three component pieces to the task.:

- **Request Clarification**
- **Feasibility Study**
- **Request Approval**

REQUEST CLARIFICATION

After the project request has been approved by the organization and project guide, the next step is to investigate the request in

order to determine the specific needs of the system taking into account.

Users whose systems are able to be linked via the Local Area Network (LAN) are the primary target audience for our project inside the firm. Everything should be offered ready-made for today's busy guy. Thus, the corresponding growth of the gateway came into existence, considering the net's wide usage in day-to-day life.

3. EXISTING SYSTEM

A brute-force dictionary attack may easily compromise an existing system's CE scheme. Bellare et al. [35] offered the DupLESS system as a solution to this issue; under this approach, the user acquires the key from a dedicated keyserver via an oblivious PRF (OPRF) protocol. The fingerprint is "blinded" by using the OPRF method. The key server has a public/private key pair that is set system-wide based on the RSA technique. So, even without knowing the original fingerprint, the key server may still return the MLE key. One defense against brute-force assaults is the use of rate-limits in key-generation requests. It would look as if the encryption key is generated from a random space if the key-server is safe. The predicate encryption approach was

expanded for use in data deduplication by Shin et al. [39].

Having said that, this approach is limited to supporting data deduplication for single users. The RCE approach, suggested by Bellare et al., has an extra tag checking mechanism [38]. The user generates the tag from the plaintext after decrypting the ciphertext, and then compares it to the matching tag. The user is asked to approve or reject the ciphertext based on whether or not the tags are consistent. Users' data is therefore assured to remain intact via the RCE program. But there are security issues with these approaches when it comes to user revocation. The convergent key gives the revoked users unauthorised access to the plaintext. This means that users' sensitive data cannot be guaranteed to remain secret.

A secure data deduplication technique was presented by Li et al. [40] to address the issue of efficient and reliable key management. This scheme employs a security Ramp secret sharing mechanism [41]. A convergent key sharing scheme and a session-key-based convergent key management system were presented by Wen et al. [36] to achieve dynamic updates in the deduplication. In order to address the issue

of data ownership changes while outsourcing, Hur et al. [21] re-encrypted the ciphertext using the group key. This ensures that only authorized users in the cloud may access the shared data. To accomplish both file-level and block-level deduplication, Chen et al. [42] suggested a system called block-level message-locked encryption (BL-MLE). In order to facilitate data access control and revocation with more flexibility, Yan et al. [43] put up a plan to deduplicate encrypted data kept in the cloud. A novel primitive named R-MLE2 was suggested by Jiang et al. [23] for cloud data deduplication with randomized tag, and it is based on static or dynamic decision trees. A secure data deduplication strategy was suggested by Liu et al. [44] using the PAKE protocol. This approach enables client-side encryption without the need for an extra independent server. Li et al. [45] presented many data deduplication algorithms that provide approved duplicate checking in a hybrid cloud architecture to tackle the permitted data deduplication issue. To address the issue of user revocation, however, these solutions mostly use re-encryption. Compute overhead is certain to be high when using the conventional re-encryption method.

The scientific community has recently become very interested in effective re-encryption algorithms. A rekeying-aware encrypted deduplication storage system was suggested by Li et al. [25] to accomplish fast re-encryption and lightweight rekeying in data deduplication. The data owner may save a lot of computational cost by using this approach to re-encrypt only a portion of the package instead of the complete thing. Also, to manage who has access to what data, the authors added ciphertext-policy attribute-based encryption [30] to REED. The REED technique, however, has a security flaw.

Disadvantages

- The Ciphertext-Policy Attribute-Based Encryption Method is not implemented by the system.
- The Bloom Filter-Based Location Selection Method is not used by an existing approach.

3.1 PROPOSED SYSTEM:

In this work, we go further into the aforementioned issues around efficient and safe re-encryption for deduplication storage. We have three main contributions: The improved encryption of the REED

system has a security flaw, as we point out [25], [26]. In other words, the so-called stub-reserved attack is something that this article proposes, and it can easily exploit this approach.

- A Bloom filter-based approach to site selection and an efficient re-encryption-based secure data deduplication strategy are presented. The revoked cloud user will be unable to access the sensitive data owned by the data owner thanks to the new location selection technique and symmetric encryption. Data privacy is therefore guaranteed. Additionally, data owners are only needed to re-encrypt a tiny portion of the package using the CAONT, rather than the full package, which significantly reduces the computation cost of the method.

The findings of our performance assessment and security analysis demonstrate that our approach is both efficient and safe.

The benefits

The suggested approach suggests a safe method for deduplicating data in the cloud that makes use of efficient re-encryption. Our plan is tailored at large organizations or user groups who want to outsource data to a distant cloud provider.

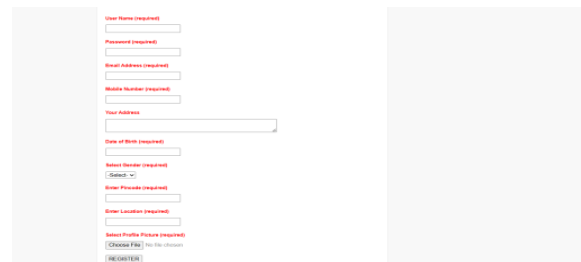
By performing deduplication on ciphertexts, the cloud service provider may significantly reduce storage overhead. User of the cloud, key server, and CSP are the three parts that make up our scheme's system

4. OUTPUT SCREENS

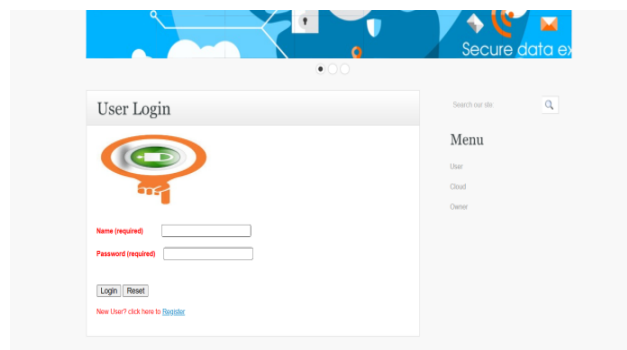
Home Page



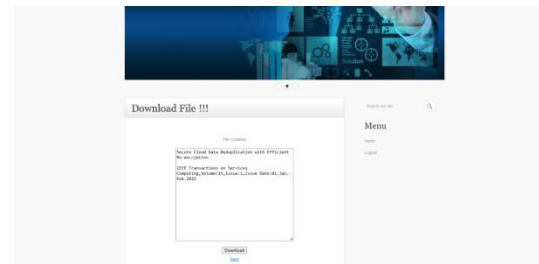
User Registration:



User Login

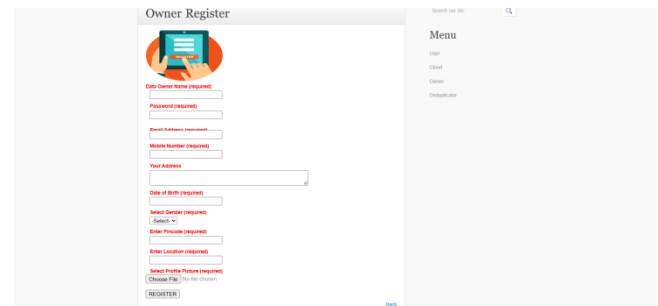
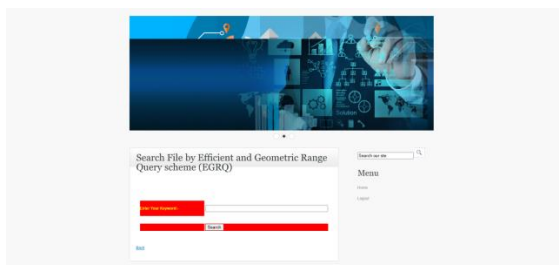


User Dashboard:



Owner Registration:

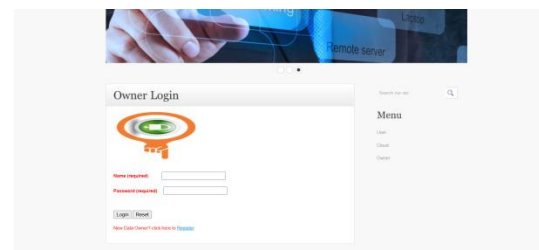
User Search File:



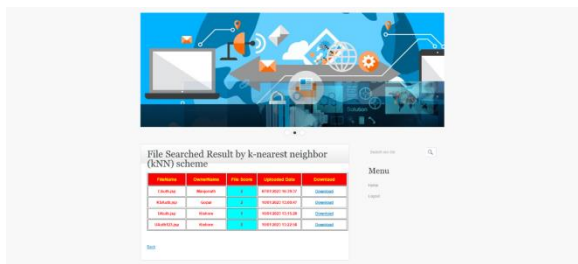
User Request Key:



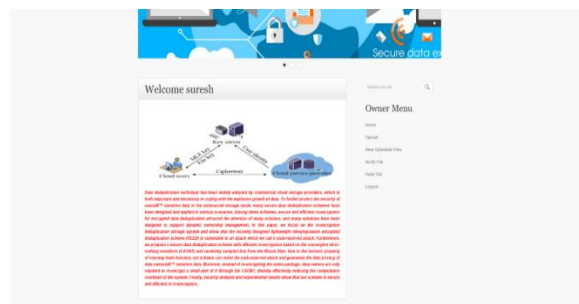
Owner Login:



User Search Result:



Owner Dashboard:

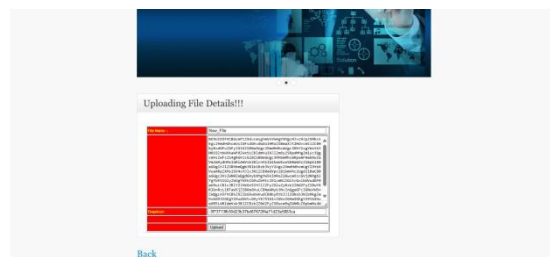


User Download:

Owner Uploaded Files:



Owner Upload :



Owner Duplication:



Owner Uploaded Successfully:



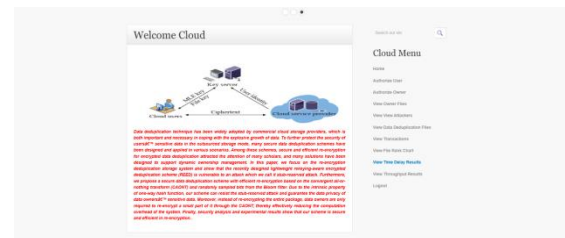
Owner Delete Files:



Cloud Server Login:



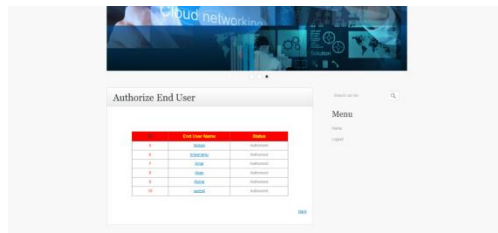
Cloud Server Dashboard:



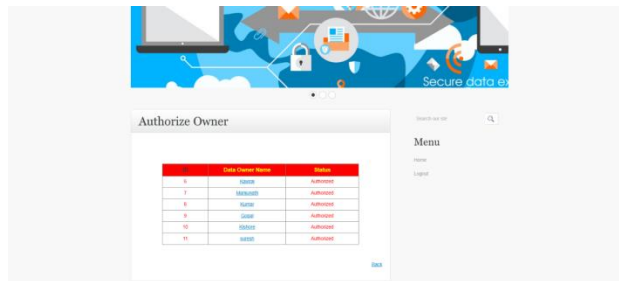
Cloud Server Files:



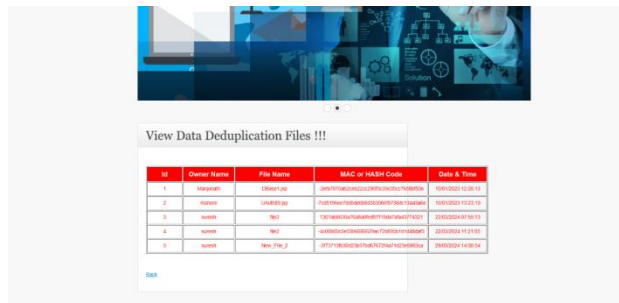
Cloud Server Authorize User:



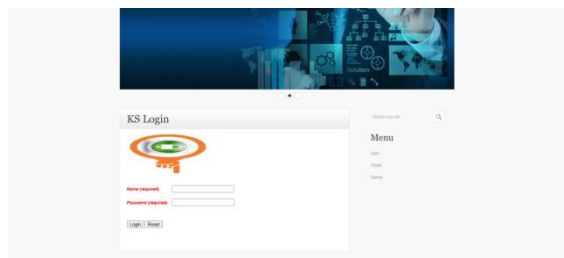
Cloud Server Authorize Owner:



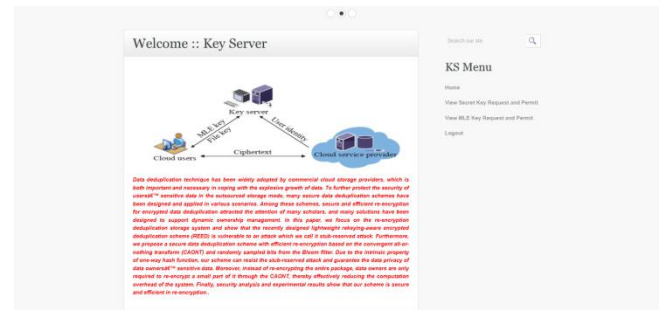
Cloud Server Duplication:



Key Server Login:



Key Server home:



Key Server Key1:



Key Server Key2:



5. CONCLUSION

Secure data de duplication with fast re-encryption and a location selection mechanism based on Bloom filters are proposed in this work. Our technique ensures the privacy of the data owners' sensitive data and is safe against the stub-reserved attack because of the intrinsic quality of one-way hash functions. And data

owners only have to re-encrypt a little portion of the package using the CAONT instead of the whole thing, which cuts down on computational overhead. Additionally, we provide comprehensive simulation testing and demonstrate that our scheme can accomplish the targeted security objectives. We found that our approach works well for re-encryption in the experiments.

6.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. Weng, and G. Pau, "Pics-on-wheels: 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/digitaluniverse/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>