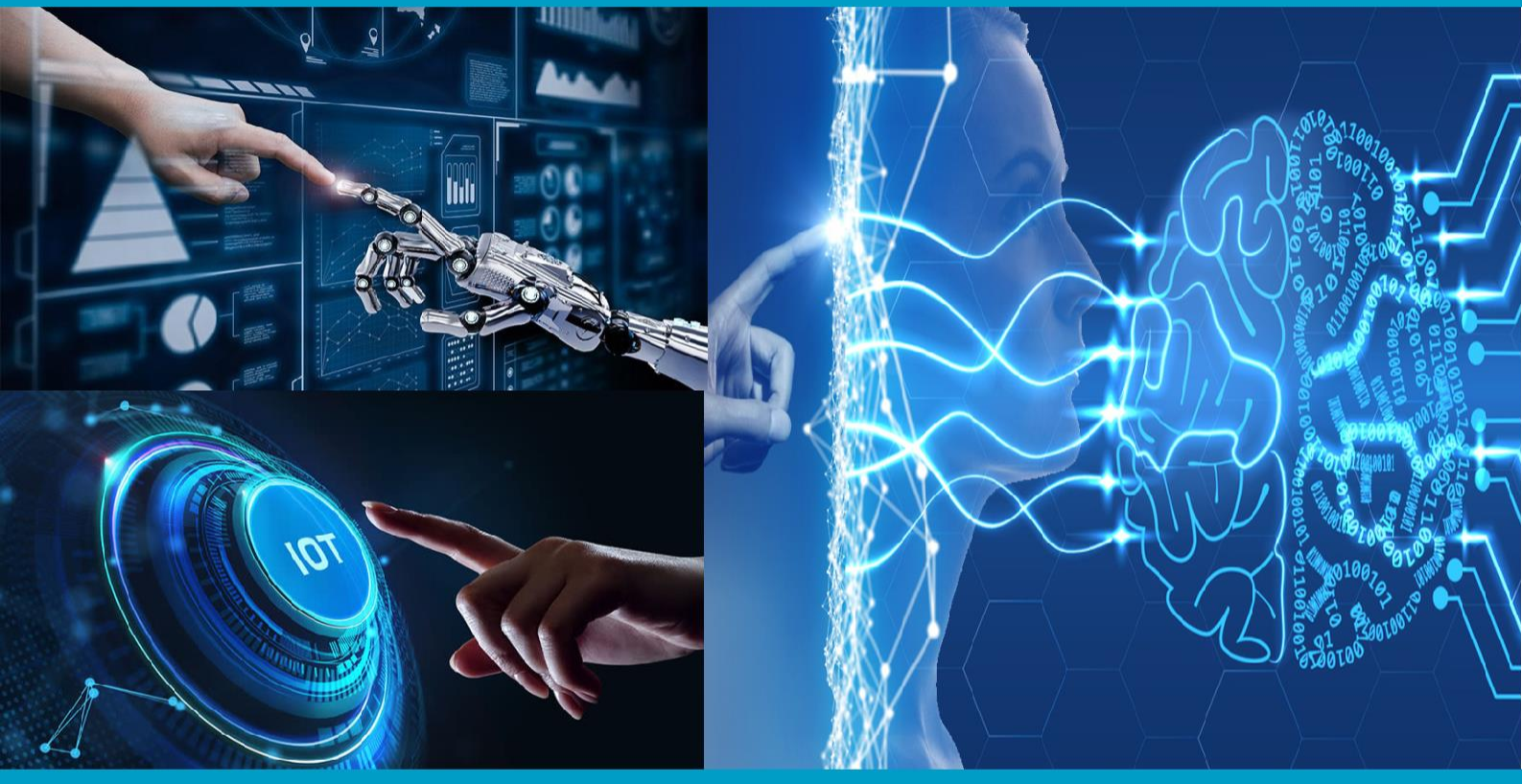
IJMRSETM

e-ISSN: 2395 - 7639




# INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

## IN SCIENCE, ENGINEERING, TECHNOLOGY AND MANAGEMENT

Volume 11, Issue 5, May 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 7.802

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

| ISSN: 2395-7639 | www.ijmrsetm.com | Impact Factor: 7.802 | A Monthly Double-Blind Peer Reviewed Journal |

| Volume 11, Issue 5, May 2024 |

# A Parallel and Scalable of Erasure Coding Support in Cloud Object Storage System

**Pon Sangeetha A [1], Rohith Vignesh G [2], Lingeswaran R [3], Mukesh D [4]**

Assistant Professor, Department of Computer Science and Engineering, Velammal Institute of Technology, Panchetti, Chennai, Tamil Nadu, India[1]

UG Scholar, Department of Computer Science and Engineering, Velammal Institute of Technology, Panchetti, Chennai, Tamil Nadu, India[2 3 4]

**ABSTRACT:** Cloud computing is a model that treats the resources on the internet as an integrated entity, cloud. Organizations proposing computing services are termed cloud providers and normally charge for their services based on their consumption. Cloud storage is an improved way out for those who wish to pay consideration to the security issues of their data. Cloud storage provides enhanced security from the occurrence of viruses. It is difficult for the information to be retrieved by any unauthenticated user since the data is encrypted when it is stored on the server. The entire server is very much secured with an innovative encryption system. The central focus of this paper is creating a protected storage system that provisions multiple tasks and this is thought-provoking when the storage system is dispersed and has no central power. Here, a proxy re-encryption scheme is suggested and combined with a distributed erasure code such that a secure and strong data storage and retrieval, but also lets a user to share his information on the cloud with a different user in the encrypted format itself. This paper facilitates the use of encoding the encrypted files and sharing files in the encrypted format itself. This paper uses the techniques of both encrypting and sharing the data. Erasure encoding supports sharing encrypted files and is valid in decentralized distributed system. A distributed erasure code is used to authorize the data safety in the dispersed cloud storage.

**KEYWORDS:** Cloud computing security, Proxy Re-Encryption, Secure storage.

## I. INTRODUCTION

Cloud storage is a virtual storage system where the information can be stored and retrieved from virtual servers instead of using physical servers and hence saving storage space. Hosting companies own large data centers. People who request to store data in the cloud storage can either purchase or rent the storage space from the cloud providers. The data center operators virtualize the resources in the back end. This is done based on the requests of the clients and given to them in the form of storage pools. The consumers can use these pools to store their documents or data objects.

Physically, the resources may be spanned across multiple servers. The users who store the information to the cloud do not need to know how the information is stored. This paper focuses on providing secure cloud storage that supports functionalities also. The cloud is measured as a large-scale distributed storage system that includes many autonomous storage servers. Data robustness is the major requirement for storage systems. There have been many schemes for keeping data over storage servers. One tactic to deliver data robustness is to duplicate a message such that each storage server provides a copy of the message. Another way is to encode the message by erasure coding. In erasure codes, the replica of the message is kept in every storage server.

In both methods, even if one of the storage server flops, the message can be recovered by any one of the remaining servers. This method is suitable for use in a distributed environment.

To provide strong confidentiality for messages in storage servers, a user can encrypt messages by a cryptographic method before applying an erasure code method to encode and store messages. There are three problems in the above straight forward integration of encryption and encoding. First, the users have the information stored in a single location. So, when many users access the data at the same time the traffic to the system becomes high. Second, the user must manage his cryptographic keys. If the user's device that is storing the cryptographic keys is lost or compromised, the security is broken. In the existing method, the owner of messages must retrieve, decode, decrypt and then forward them to another user.

To deliver strong confidentiality for message in storage servers, the idea that consists of distributed storage servers and

| ISSN: 2395-7639 | www.ijmrsetm.com | Impact Factor: 7.802 | A Monthly Double-Blind Peer Reviewed Journal |

| Volume 11, Issue 5, May 2024 |

key servers is considered. If the consumer keeps the key, there are risks of the key getting missed. Therefore, the key is kept in the key servers where it is partly encoded. The key servers must be encrypted to guarantee security and they should serve as an autonomous different from the storage servers. With this concern, a novel decentralized erasure code is suggested, appropriate for usage in a distributed storage system.

The consumer will upload the files and it will be encrypted with AES Encryption and Proxy re-encryption. The data will be divided into small pieces by using a dividing key inside the cloud storage. The data is stored in different storage locations which will be monitored by the unique data distributors. If the valid user is accessing the data, it is retrieved in reversible manner from the cloud storage. The use of encryption and erasure coding helps to meet the requirements of data security, data confidentiality, and information sharing.

## II. LITERATURE REVIEW

Literature research is the most important step in the software development process. Before creating a tool, it is important to determine the time factor, profitability, and company strengths. With these in place, the next 10 steps are to decide which operating systems and languages you can use to develop your tools. Once programmers start building tools, they need a lot of external support. This support can come from experienced programmers, books, or websites. The above evaluations will be considered in the development of the proposed system before building the system.

**Peng Zeng, Kim-Kwang Raymond Choo, "A New Kind of Conditional Proxy Re-Encryption for Secure Cloud Storage",2018**

Secure cloud storage has important applications in our big data-driven society, and to achieve secure cloud storage we need to enforce strong access control mechanisms. Proxy re-encryption (PRE) has been shown to be an effective tool for constructing cryptographically enforced access control schemes. In a traditional PRE scheme, a semi-trusted proxy can convert all ciphertexts for a delegator to ciphertexts for a delegate once the proxy obtains the relevant re-encryption key from the delegator. In many practical applications, however, a fine-grained delegation of decryption abilities may be demanded and thus the notion of conditional PRE (C-PRE) is introduced, which allows only the ciphertexts satisfying a concrete condition to be converted by the proxy. In this paper, we introduce a special kind of C-PRE, sender-specified PRE (SSPRE), which enables the delegator to delegate the decryption right of the ciphertexts from a specified sender to his/her delegate. We give a formal definition of SS-PRE and its security model. We also provide concrete constructions of an IND-CPA secure SS-PRE scheme and an IND-CCA secure SS-PRE scheme with the properties of unidirectionality and single-use and prove the security of both schemes in the standard model. The detailed analysis shows that our new IND-CCA secure SS-PRE scheme achieves a higher efficiency in computation cost and ciphertext size than conventional C-PRE schemes

**Wei-Hao Chen, Chun-I Fan, Yi-Fan Tseng, "Efficient Key-Aggregate Proxy Re-Encryption for Secure Data Sharing in Clouds", 2018**

A. Protecting sensitive files stored in the clouds from being accessed by malicious attackers is essential to the success of the clouds. In proxy re-encryption schemes, users delegate their encrypted files to other users by using re-encryption keys, which elegantly transfers the users' burden to the cloud servers. Moreover, one can adopt conditional proxy re-encryption schemes to employ their access control policy on the files to be shared. However, we recognize that the size of re-encryption keys will grow linearly with the number of the condition values, which may be impractical in low computational devices. In this paper, we combine a key-aggregate approach and a proxy re-encryption scheme into a key-aggregate proxy re-encryption scheme. It is worth mentioning that the proposed scheme is the first key-aggregate proxy re-encryption scheme. As a side note, the size of re-encryption keys is constant.

**Yoshiko Yasumura, Hiroki Imabayashi, Hayato Yamana" Attribute-based Proxy Re-encryption Method for Revocation in Cloud Storage: Reduction of Communication Cost at Re-encryption", 2018**

Attribute-based encryption (ABE) enables both data security and access control by defining users with attributes so that only those users who have matching attributes can decrypt them. For real-world applications of ABE, revocation of users or their attributes is necessary so that revoked users can no longer decrypt the data. In actual implementations, ABE is used in hybrid with a symmetric encryption scheme such as the advanced encryption standard (AES) where data is encrypted with AES and the AES key is encrypted with ABE. The hybrid encryption scheme requires re-encryption of the data upon revocation to ensure that the revoked users can no longer decrypt that data. To re-encrypt the data, the data owner (DO) must download the data from the cloud, then decrypt, encrypt, and upload the data back to the cloud, resulting in both huge communication costs and computational burden on the DO depending on the size of the data to be re-encrypted. In this paper, we propose an attribute-based proxy re-encryption method in which data can

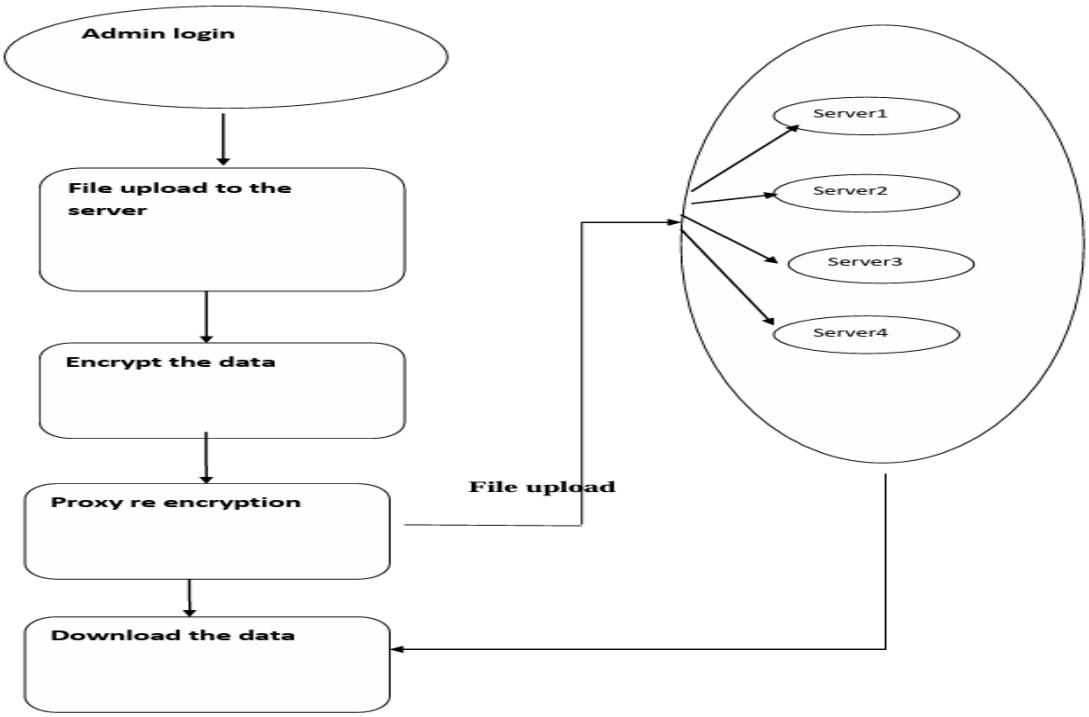
| ISSN: 2395-7639 | www.ijmrsetm.com | Impact Factor: 7.802 | A Monthly Double-Blind Peer Reviewed Journal |

| Volume 11, Issue 5, May 2024 |

be re-encrypted in the cloud without downloading any data by adopting both ABE and Syalim's encryption scheme. Our proposed scheme reduces the communication cost between the DO and cloud storage. Experimental results show that the proposed method reduces the communication cost by as much as one quarter compared to that of the trivial solution.

## III. METHODOLOGY

The system architect establishes the basic structure of the system, The user creates a login for uploading his or her file to the cloud. After logging into the account, the user browses for the file to be uploaded and uploads the file to the cloud. The file gets encrypted using AES algorithm and gets split into four parts. This is done using Erasure code technique. The encrypted file is again encoded using MD5 algorithm which produces a 32-bit hexadecimal key. The key gets stored in the server. The user selects the file that is to be downloaded and decryption of the file happens. If the file must be shared, the file is encrypted using AES. The file can be saved to the user's system and is shared with the receiver. The receiver logs into the account and decrypts the received file. This restricts the breach of security while the file is transferred. The whole functionality of the proposed idea is provided in Fig. 2. AES (Advanced Encryption Standard) is used to encrypt the file that is uploaded. This algorithm is until today proved to be secure and is practically not hacked yet. In this paper, we use a 128-bit key since it contains a smaller number of rounds (10 rounds) and hence the processing is easy and consumes less time. Erasure coding is used to split and save the file in the cloud so that when a hacker tries to retrieve a file, he cannot retrieve the file completely and even part of the file remains encrypted. Proxy re-encryption is used to encode each part of the file again and this results in a 32-bit code which is stored in the server.




## IV. RESULTS AND DISCUSSION

The objective for securing group data sharing in the cloud using an untraceable digital data store approach is to develop a robust system that ensures the confidentiality and integrity of shared data among group members. The goal is to prevent unauthorized access and maintain anonymity, making it difficult for external entities to trace or compromise sensitive information. The storage system is allocated by different data containers. Once owner uploads the data with AES encryption mechanism, system again takes the data and makes Secure Data segregation process.

### ALGORITHM

### Advanced Encryption Standard

The Advanced Encryption Standard (AES) is a widely used symmetric key algorithm for secure data transmission and storage. It operates on fixed-size blocks of data (typically 128 bits or 16 bytes) and employs a substitution-permutation

| ISSN: 2395-7639 | www.ijmrsetm.com | Impact Factor: 7.802 | A Monthly Double-Blind Peer Reviewed Journal |

| Volume 11, Issue 5, May 2024 |

network (SPN) structure. AES supports key lengths of 128, 192, or 256 bits. During encryption, data undergoes several rounds, each involving steps like SubBytes (substitution), ShiftRows (permutation), MixColumns (transformation), and Add Round Key (XOR with the round key). The number of rounds depends on the key length: 10 rounds for a 128-bit key, 12 rounds for a 192-bit key, and 14 rounds for a 256-bit key. AES is designed to be secure, efficient, and resistant to known attacks, making it a cornerstone of modern cryptography.

**MD5 Hashing Algorithm**

The Md5 algorithm is a hash function that produces 128-bit hash value that is stored as a 32-bit hexadecimal value. The advantage of MD5 is it produces different hash values for the same plain text. While MD5 was once widely used and considered secure, its vulnerabilities have made it unsuitable for cryptographic purposes where collision resistance is essential. It's still used in some non-security-critical applications, but caution should be exercised when relying on MD5 for any cryptographic tasks.

**Erasure Code Technique**

Erasure code is a form of forward error correction (FEC) code. This technique converts a message of 't' symbols into an elongated message with 'p' symbols such that the initial plain text message can be obtained from a subgroup of the 'p' symbols. The component s = t/p is named as the code rate, the component t'/t, where t' represents the number of symbols essential for recovery which is called the reception efficiency. Erasure coding is a fundamental technique for providing fault tolerance and data integrity in distributed storage and communication systems, offering efficient storage utilization and resilience against failures. The encoding process involves generating parity data from the original data using the chosen erasure code technique. The decoding process reconstructs lost or corrupted data using the available data and parity information. Depending on the specific technique used, decoding may require accessing a subset of the available data and parity.

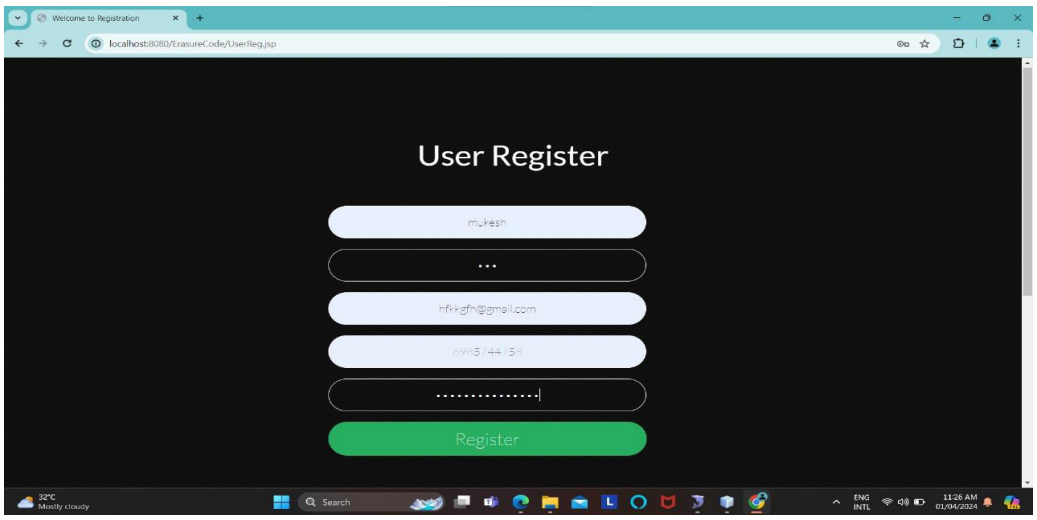
**Proxy Re-encryption**

Proxy re-encryption schemes are crypto frameworks that allow intermediaries (proxies) to alter a cipher content which has been encoded for one client, so that it may be unscrambled by another client. By using proxy re-encryption technique, the encrypted data (cipher text) in the cloud is again altered by the user. It provides highly secured information stored in the cloud. Every user will have a public key and private key. The public key of each user is well-known to everybody, but the private key is known only by the user. This is achieved using MD5 algorithm. Keeping in mind the end goal to preserve security, the customers will encode their information when they out-source it to the cloud. In any case, the encrypted type of files significantly impedes the usage due to its haphazardness. Numerous research has been done for the goal of information usage with functionalities without compromising information security. Fig. 4 explains the pseudo code of proxy re-encryption. Homomorphism: Given two cipher texts c and d on plaintexts m and n separately, an individual can acquire the cipher text on the original message m + n or m*n by calculating c and d without the necessity to decrypt cipher texts. Proxy re-encryption: Proxy re-encryption is the process where the data that is already encrypted by a certain encrypting algorithm is again encoded using a hashing algorithm. This is being done to improve the security of the stored files. Threshold decryption: By distributing the private key into numerous fragments of undisclosed portions, all clients can work collectively to obtain the original plain text message which serves as the outcome of the function. Erasure codes are used in the method of threshold proxy re-encryption. Decentralized erasure code procedure can be used for decentralized storage system which offers secure cloud storage by using unconnected servers for storage and key.

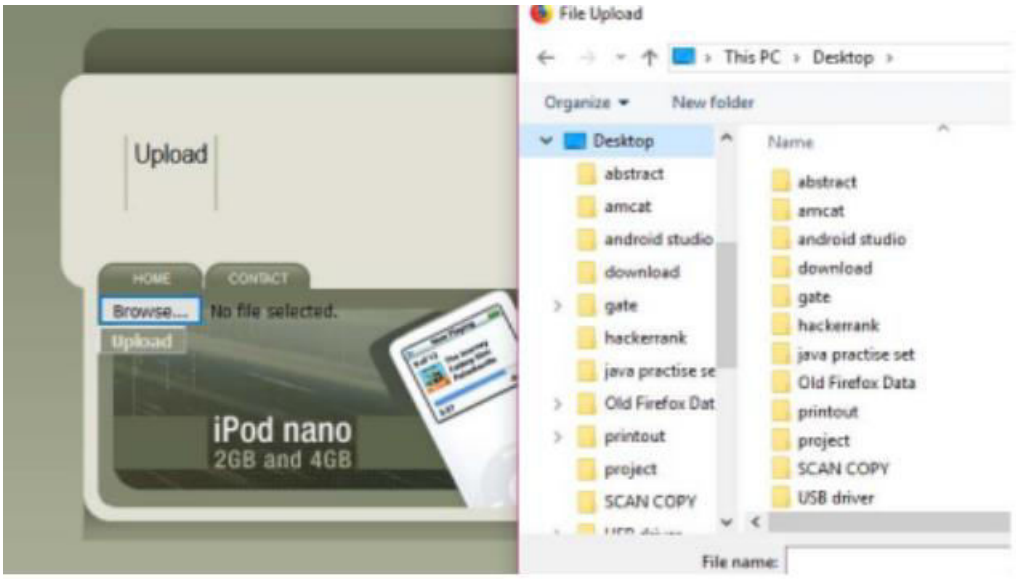
**OUTPUT:**

| ISSN: 2395-7639 | www.ijmrsetm.com | Impact Factor: 7.802 | A Monthly Double-Blind Peer Reviewed Journal |

| Volume 11, Issue 5, May 2024 |

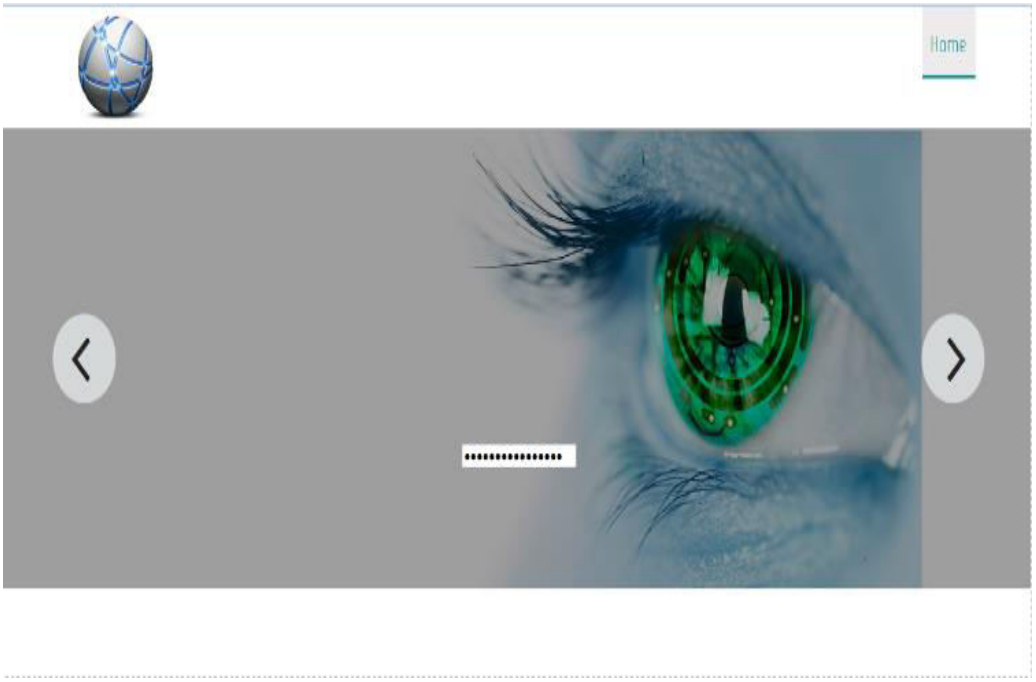






| ISSN: 2395-7639 | www.ijmrsetm.com | Impact Factor: 7.802 | A Monthly Double-Blind Peer Reviewed Journal |

| Volume 11, Issue 5, May 2024 |

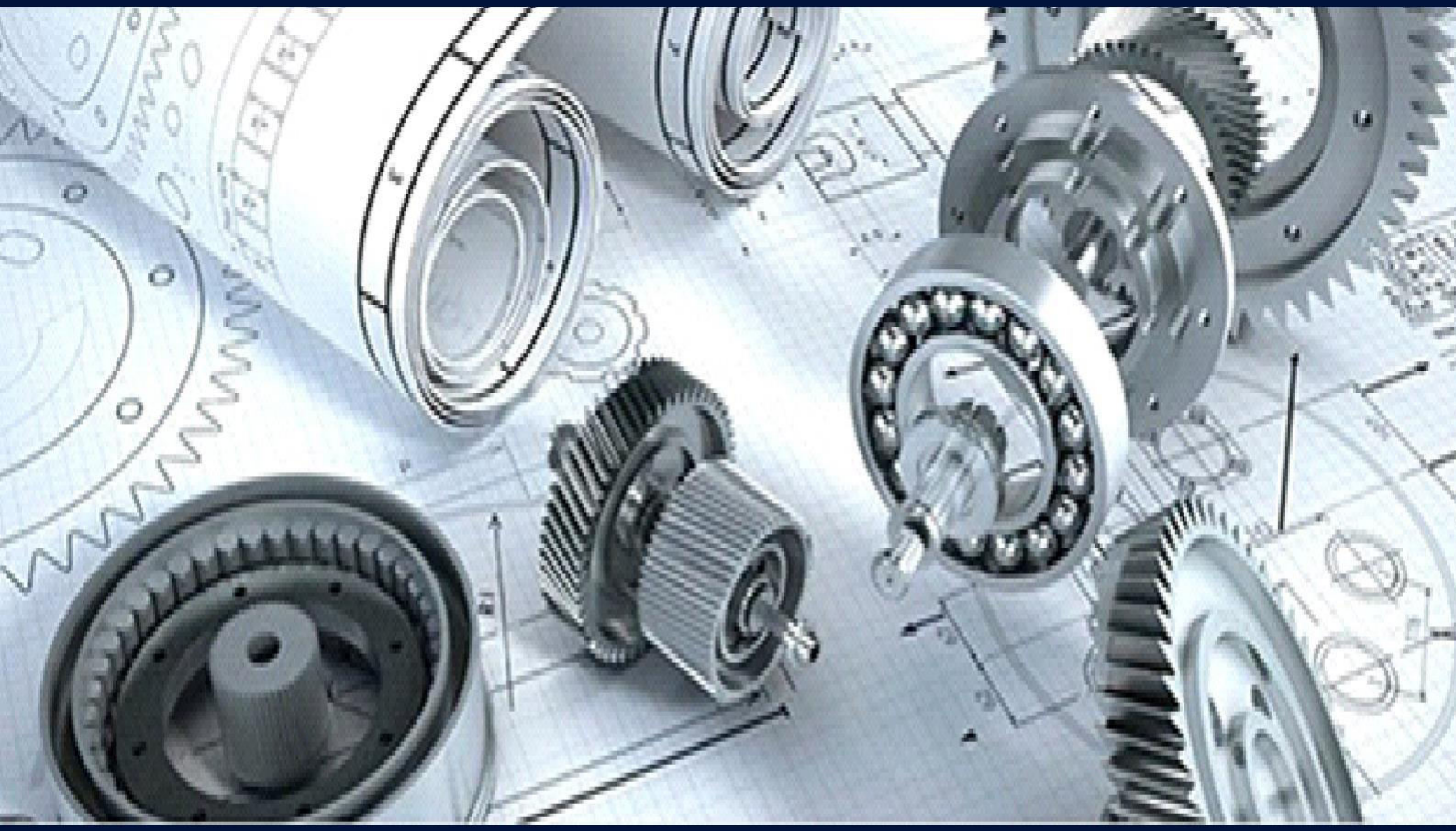



## V. CONCLUSION

In conclusion, a protected cloud storage framework that supports functionalities is considered. We incorporate a novel proposal which is the proxy re-encryption scheme and erasure codes over encrypted messages. The proxy re-encryption framework supports encoding, forwarding, and decryption functions in a decentralized manner. Proxy re-encryption is used to re-encrypt the data that is already encrypted, and this reduces the storage space when it is being stored in a distributed environment. Additionally, each storage server individually implements encoding and re-encryption, and every key server autonomously carries out partial decryption. The storage system and freshly proposed file system are highly harmonious and can provide a new level of security.

## REFERENCES

1. Xuechen Zhang ECE Department Wayne State Universities Trans. Kei Davison Alamos National Laboratory Parallel and Distributed Systems, vol. 21, no. 11, pp. 1586-1594, Nov. 2010.
2. Prof.dr.ir. H.J. Sips Technische Universities Delft, promoter Prof.dr.ir. A.J.C. van Gemund Technische Universities Delft Prof.dr.ir. H.E. Bal. 7 December
3. Nihar B. Shah, K. V. Rashmi, Kennan Ramchandran, Fellow, IEEE, and P. Vijay Kumar, Fellow, IEEE. 2011.
4. Babak Behzad, Surendra Byna, Prabhat Lawrence Berkeley National Laboratory.2011 IEEE.
5. Xiaowen Chu, Chongjin Liu, Kai Ouyang, Ling Sing Yung, Hai Liu. Hong Kong. 2010 IEEE.
6. Huayu Zhang, Hui Li, Bing Zhu, Jun Chen 2014 IEEE 33rd International Symposium on Reliable Distributed System.
7. Weidong Sun, Yijie Wang, Yongquan Fu, Xiaoqiang Pei 2014 IEEE 8th International Symposium on Service Oriented System Engineering.
8. Jibin Wang, Lili Yang, Hu Zhang, Zhaogang Xu, Ying Guo 2015 Third International Conference on Advanced Cloud and Big Data.
9. David Nunez, Isaac Agudo, Javier Lopez 2015 IEEE 28th Computer Security Foundations Symposium.
10. Chungsik Song, Younghee Park, Jerry Gao, Sri Kinnera Zegers 2015 IEEE First International Conference on Big Data Computing Service and Applications (Big Data Service) (2015)
11. Jatinder Paul Singh, Shobhit University, Meerut, India Mamta Shobhit University, Meerut, India, Sunil Kumar, IIMT Engg. College, India 2015 National Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM).
12. Natazul haque Sultan, Ferdous Ahmed Barbhuiya, 2016 IEEE World Congress on Services (SERVICES) (2016) San Francisco, CA, USA June 27, 2016 to July 2, 2016.



INTERNATIONAL STANDARD SERIAL NUMBER INDIA

# INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH

## IN SCIENCE, ENGINEERING, TECHNOLOGY AND MANAGEMENT

+91 99405 72462 +91 63819 07438 ijmrsetm@gmail.com

www.ijmrsetm.com