International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 11s
DOI: https://doi.org/10.17762/ijritcc.v11i11s.8084
***Article Received:** 26 June 2023 **Revised:** 24 August 2023 **Accepted:** 04 September 2023*

Implementation of Dynamic Virtual Cloud Architecture for Privacy Data Storage

Banoth Anantharam[1], Dr.Neeraj Sharma[2] Dr. B.Kavitha Rani [3]
[1]Research Scholar - Department of Computer Science and Engineering, School of Engineering, Sri Satya Sai University of Technology & Medical Sciences, Sehore,MP,India.
Emailid: ananth502.ram@gmail.com
[2]Department of Computer Science and Engineering, School of Engineering, Sri Satya Sai University of Technology & Medical Sciences, Sehore,MP,India.
[3]Department of COMPUTER SCIENCE & ENGINEERING, CMR Technical Campus, Hyderabad, Hyderabad, Telangana 501401.

Abstract: Nowadays rapidly developing technologies, cloud computing offers versatile services. However, cloud computing presents a challenge to secure information sharing. Customers can securely share their data with others and remotely store it in the cloud using cloud storage services. In recent times, cloud storage typically represents as the primary method of external data storage. The primary challenge is safeguarding the cloud-based data against attacks. Over the information network, the growth of private or semi-private information has increased. The search techniques have not been addressed by privacy safeguards. As there is no suitable audit system, the validity of the stored data has become in question. In addition, user authentication presents additional difficulties. Hence in order to solve these issues, Design and implementation of dynamic virtual cloud architecture for privacy data storage is presented. In this approach, third-party audits are presented accompanied a new, regenerative public audit methodology. A distributed KDC (Key Distribution Center) is employed to encrypt the data. Documents can be stored on a private server in plain word form, which compromise the protection of privacy. As a result, system security can be improved to make the documents safer and more effective. The main objective of this Virtual Cloud Architecture is to achieve data confidentiality, as well as authenticity.

Keywords: Cloud Computing, Data Privacy, Cloud Storage, Virtual Cloud Architecture.

I. INTRODUCTION

A large-scale distributed and virtual machine computing infrastructure, cloud computing has rapidly developed during the last years. Due to expanding network capacity and connections that are both dependable and adaptable, customers can now subscribe to high-quality services using data and software that are only available in remote data centers. The companies are store data on cloud storage for file archiving, backup, and even primary storage due to the increased flexibility and savings in budgets.

Since it currently offers a flexible on-demand data outsourcing service with attractive advantages, cloud storage is gaining popularity because it eliminates the need to purchase costly hardware, software, and individualized maintenance costs. With its many advantages, cloud storage is attracting more and more businesses, individuals to move their data from local to remote cloud servers [4]. It offers consumers an on-demand data outsourcing service model.

In the cloud storage model, data is stored in virtualized, distributed storage groups that are acquired by third-party persons. Google and Amazon, among others, own large data centers that are available for lease or purchase by anyone. The owners of the data are mostly free of issues regarding security, dependability, and availability by storing it in a third-party cloud. Cloud storage service providers offer virtualized resources to store user files or data based on the requirements of the user. Although physically it may be using multiple servers, the user may have the feel that they are only using single storage [1].

Security measures are necessary because shared data from cloud servers frequently contains confidential or sensitive customer information. The trustworthiness of the data should additionally be checked. The privacy of shared records in the cloud is difficult to maintain, particularly in a hybrid cloud and big data environment. However, big data has a significant processing demand due to its excessive range, volume, and veracity of records. Using cloud storage services, users can store their data remotely in the cloud [2]. The sensitive data stored in the public cloud by privileged users, such as corporate companies and government agencies are highly vulnerable in the hands of cloud providers and hackers. Using a cloud server, it's hard to keep all important data safe across all client applications.

Both cloud services and other computing systems are susceptible to mistakes, failures, and attacks. The main differentiation is that a cloud's vulnerabilities or attack might have a significant impact on a large number of users and customers, raising serious security concerns. In order to ensure

***Article Received:** 26 June 2023 **Revised:** 24 August 2023 **Accepted:** 04 September 2023*

the successful implementation of cloud-based services, cloud security focuses not only on preventing unauthorized access to cloud data by its owners but also on maintaining data integrity and confidentiality. A service provider should be ensured that user data are effectively protected and can only be accessed by authorized and authenticated users. It is possible for Cloud storage providers to display confidential or private information because users can store certain personal or confidential information without knowing where it is stored. It becomes more challenging for data owners to ensure data integrity or reliability since some storage nodes in a cloud system might not be trustworthy. If the outsourced data integrity is a requirement for these networked storage systems, data repair bandwidth could be reduced using an alternate approach called regenerating coding Therefore, regenerating coding-based cloud storage systems requires the enabling of remote data integrity verification [3].

Because it can ensure that the outsourced data on the cloud server are not tampered by attackers. The public auditing protocol is essential to the success of cloud computing. Because of its significance, public reviewing protocol has improved extensive consideration in the previous year’s [5]. In most cases, an independent third party is involved in the process of locating evidence through a number of methods, such as investigation, physical inspection, confirmation, observation, etc. Any organization's cloud implementation audit requires monitoring all internal and external processes to identify compliance requirements, such as Service Level Agreements (SLAs), laws, and corporate policies.

Existing solutions do not meet these requirements for learning new information. The implementation of storage complexity is less guaranteed by some systems: although it shouldn't be essentially constant exact knowledge, the server should save at least as much knowledge as the customer. Also, all previous techniques required full access to the file from the server, which is impossible to handle once a lot of technical issues have been resolved. Hence there is a need to design an effective cloud environment for data privacy and integrity.

This work, presents Implementation of dynamic virtual cloud architecture for privacy data storage. The following is the structure of the remaining work: The literature survey is described in section II. In this section III demonstrates the implementation of dynamic virtual cloud architecture for privacy data storage. The section IV evaluates the result analysis of presented implementation. In section V, the work is finally concluded.

II. LITERATURE SURVEY

Wei Liang, Yongkai Fan, Kuan-Ching Li, Dafang Zhang, and Jean-Luc Gaudiot et.al. [7] explains how to store and retrieve data safely in industrial blockchain network environments. With increasing storage systems decentration, tamper-proof, real-time monitoring, management, for the blockchain-based network, a secure data storage and recovery scheme is proposed. This design supports the dynamic storage, efficient repair, and update of distributed data in industrial nodes data storage systems. Data is repaired and stored between failed nodes using a local regenerative code technology that protects user data. This approach appears to be promising with good security and real-time performance, as shown by testing results that demonstrate it increases data storage rate by 8.6% while also increasing multinode data repair rate by 9%.

Miss. Nirupamashree, Mrs. Pushpa R et. al. [10] provides A Regenerating Code-Based Secured Public Auditing for Cloud Storage. For the regenerating-code-based cloud storage, they provide a public auditing system. In the traditional public auditing system model, to address the regeneration problem in the absence of data owners, they introduce a proxy with the authority to regenerate failed authenticators. In addition, they develop a novel public verifiable authenticator that can be regenerated with partial keys and is generated by a few keys. As a result, this approach could entirely relieve data owners of their online burden. To protect the privacy of the data, they also use a pseudorandom function to randomly select the encode coefficients. This system's excellent performance and ability to be successfully integrated into the regenerating -code-based cloud storage are shown through experimental evaluation. This system is random oracle model-provably secure, according to a comprehensive security analysis.

Pritha, K. and Nivethitha, M et. al. [12] presents Regenerated Code for Secure Cloud Sharing. In this study, a public auditing technique for cloud storage that uses regenerating code is presented. In the absence of data owners, this approach resolves the regeneration problem. The users request that the data's integrity be checked by a Third Party Auditor (TPA). Additionally, the AES (Advanced Encryption Standard) algorithm is used to encrypt the data in order to protected their privacy. Due to its high level of security, this technique can be easily included into cloud storage that uses regenerating codes. The data is effectively protected using the MD5 (Message Digest rule) method.

P. Balasermathi and Mrs.S.G. Sandhya et. al., [14] discusses a security study on the integrity of data in regenerating cloud storage based on coding. Outsourced data stored in cloud storage can be protected against corruption using a data integrity protection method. A mobile Byzantine adversarial model was used to develop , developing it simple for a client to check the integrity of random subsets of outsourced data for malicious or generic corruptions. A flexible method for auditing the integrity of distributed storage that makes use of distributed erasure-coded data and homomorphic tokens. With relatively limited communication and computation costs, the proposed design enables users to audit the cloud storage. The auditing result not only achieves fast data error localization but also a strong ensure of correctness for cloud storage. Byzantine failure, malicious data modification attacks, and even server collusion attacks can't break this strategy's resilience.

Jing Chen, Ruiying Du, Yuling Peng, Quan Yuan and Minghui Zheng et. al. [16] discusses Effective Remote Data Checking and Fixing in Cloud Storage Based on Regenerating Codes. Based on the codes that regenerate with the least amount of bandwidth, they show an effective Remote Data Checking and Repairing (RDCR) method. By allowing a third party to carry out the public integrity verification, owners of data will find it easier to check their data's integrity with this scheme. Additionally, unlike earlier schemes, our scheme provides accurate correction of corrupted data, substantially reducing computation costs. They developed this scheme, and the experiment results reveal that RDCR has lower computing overhead and communication cost than existing schemes.

Dongju Yang, Chuan Ren et. al. [18] explains VCSS: A Framework for Integrating Open Cloud Storage Services. Virtual Framework for Cloud Storage Service (VCSS) integrates various open, standard Cloud Storage Services to establish a consistent pool of virtual storage resources. A service metadata model with rich NFP (Non-Functional Properties) to determine and select the best services to store and replicate the file, a service scheduling model is utilized. A service repository acts as a virtual storage resource pool that enables resources to be physically distributed stored and conceptually centralized controlled. The initial demonstrative use is promising and shows that VCSS is capable of achieving predicted effectiveness.

III. IMPLEMENTATION OF DYNAMIC VIRTUAL CLOUD ARCHITECTURE

The authenticity, availability, and integrity of the data at danger since data owners have no actual authority over the events that occur to their outsourced data. Consequently, cloud service providers may behave dishonestly by hiding data loss or corruption and maintaining that the files are still appropriately saved in the cloud for monetary or reputational reasons. Thus, it makes perfect sense for users to put in place a reliable protocol to carry out periodic inspections on their outsourced data to make sure the cloud actually preserves their data accurately. So, this part presents the implementation of a dynamic virtual cloud environment for the storing of personal data in order to address these issues. The Block diagram of presented approach is shown in Fig. 1.

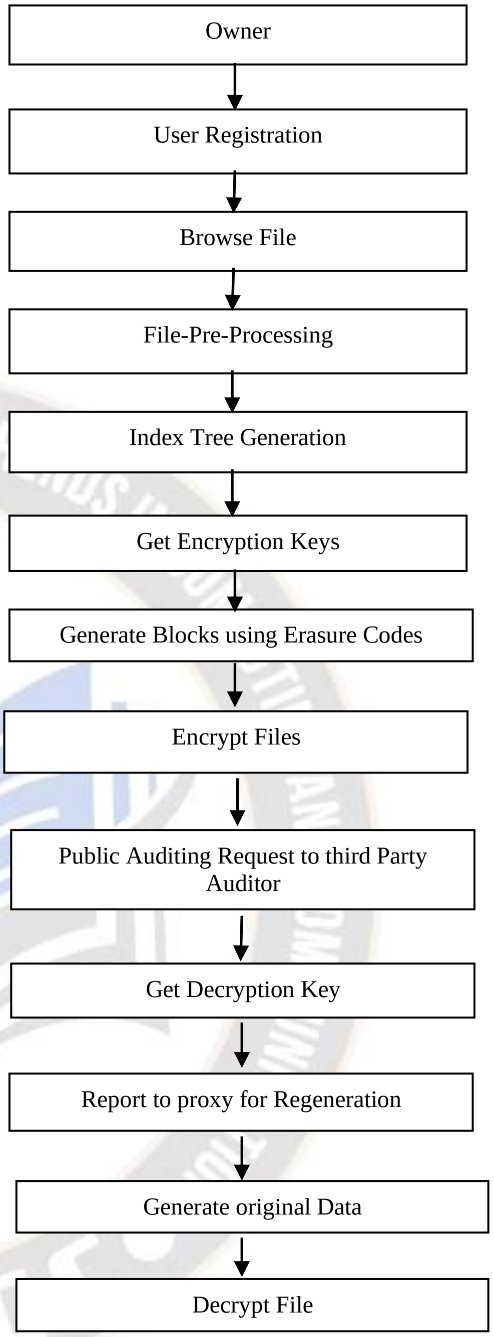

Fig. 1: Block Diagram of Dynamic Virtual Cloud Architecture

Any individual using a CSP's (Cloud Service Providers) cloud services for a Cloud Customer based on a relationship between that Cloud Customer and the Cloud User is the owner. The user registration process requires some details User Name, Email Address, Password, etc. The generic security role that the newly created user profile needs to access the cloud systems are assigned during New User Registration. A user who has previously registered for a cloud website or program is referred to as a registered user.

DOI: https://doi.org/10.17762/ijritcc.v11i11s.8084
Article Received: *26 June 2023* ***Revised:*** *24 August 2023* ***Accepted:*** *04 September 2023*

In order to prove their identity, registered users typically provide the system with some kind of credentials, such as a username or e-mail address and a password: This is called "logging in." Digital information is maintained in logical pools, or "the cloud," under the cloud storage model of computer data storage. A hosting company often manages and controls the physical environment, the physical storage spans multiple servers. Files, business data, videos, and images are saved by Cloud Storage on remote servers. Through an internet connection, users send data to servers, where it is saved in a virtual machine on a physical server. The cloud storage provides different applications to user which are as follows: i) File storage: Files, emails, and other sorts of data can all be stored by the user on the cloud; ii) File sharing: Using the cloud makes it simple to exchange files with multiple users at same time; iii) Backing up data: The owner can also use the cloud to protect his/her files.

The owner stores their files, images, etc. on the cloud. Whenever they want a particular file, he/she browses from their data which is stored in the cloud. The text of a source file is manipulated during pre-processing, usually as the first phase of translation initiated by a compiler invocation. Macro substitution, checking for conditional compilation directives, and files are all common pre-processing tasks.

Data Owner has a list of files that have been cloud-specified and cloud-encoded. The user has the ability to search this encrypted data. On this system, the data owner makes a safe index trees, and then the encrypted file is produced. It generates the index trees. The index tree's input: The identifiers $FID = \{FID$. are associated with the document collection . T is the index tree output. The encrypted files and index tree are saved on the cloud's server. The distribution of keys necessary for file decryption shall be the responsibility of the data owner to the authorised users. The cloud server provides the user with encrypted top-k results after searching the index tree in response to a document query request. With the secret key that the data owner provided, decrypt the files that were retrieved.

Users can store their respected hash and encrypted file blocks on a cloud server. This file block encryption uses a distributed KDC (Key Distribution Center). The system uses distributed KDCs because if one is busy, another will be used. Performance is enhanced and the load is distributed to KDC. The following is the generation of keys:

$Key\ Generation\ PK = \{pk1, pk2, pk3, \ldots, pkn\} (1)$

PK indicates for the collection of generated public keys.

$SK = \{sk1, sk2, sk3, \ldots skn\} (2)$

The set of generated private keys that are related to the public key is called SK. Firstly a random generator and random integrator is chosen and sets $g_1 = g^x$. Two hash functions select a random Public Key Generation (PKG). $PK = (g, g_1, g_2, H_1, H_2)$ and the master key $MK = x$ are the final outputs.

The PKG terminates the key generation algorithm if the search identity ID private key requests for each user's identification have already been registered. The file blocks can be encrypted with the key. Before storing block files on cloud storage, the user generates and stores the hash on the server. A user submits the extended query to the database owner when they want to demonstrate their identity. The public cloud user Query is given as

$Q = \{Q_1, Q_2, \ldots Q_n\}$ (3)

Where, Q is the collection of all public cloud queries. On the private server, user authentication is provided as

$U = \{U_1, U_2, \ldots U_n\}$ (4)

Where U indicates the total number of authenticated private server users. The generation and distribution of tokens are provided as

$T = \{T_1, T_2, \ldots T_n\}$ (5)

Where, T is the collection of tokens that the private server generates for its authenticated users. At KDC, key generation

$K = \{K_1, K_2, \ldots K_n\}$ (6)

Where, K is the set of all keys stored at KDC and utilized by the user for data decryption. Ranking tokens and decrypting data

$R = \{R_1, R_2, \ldots R_n\}$ (7)

Where, R represents the collection of all ranked responses to a specific input query. For file block check for completeness, the user can request a cloud server store in TPA (Third Party Authority). TPA keeps the hash blocks in storage. It requires a user to verify the validity of specific file requests. It compares the file block hash it obtained from its database with the hash store. If the hash matches, it will let the user know that the server's store files are not corrupted. TPA sends a proxy request to fix a corrupt file. A coded proxy is used for regeneration.

Assuming that, the ciphertext CT was encrypted under the user's ID for some Time (Ti) and that the user has a private key (SK). The original data is decrypted in the decryption phase. For decryption, Data owner enters his decryption key, if it matches then it decrypts the data and original data is given to user. If an intruder is trying to get the information and his decryption key is not matches then it reports the proxy server for regeneration. This regeneration code enables Proxy to restore damaged server files. The file is then once again checked for recovery by TPA. TPA retrieves the original file from the database and notifies the user that the file has been recovered in the event that the user file is damaged by the intruder. The regenerate code is sent to data owner for getting

DOI: https://doi.org/10.17762/ijritcc.v11i11s.8084
***Article Received:** 26 June 2023 **Revised:** 24 August 2023 **Accepted:** 04 September 2023*

his/her original data. Hence in this manner, this approach provides authentication and privacy to user data.

IV. RESULT ANALYSIS

In this section, Implementation of dynamic virtual cloud environment for privacy data storage is presented. The result analysis of presented implementation is evaluated here. The public datasets offered by Cloud Storage can be accessed by the community and integrated into applications. Google provides public access to these datasets through tools like the Google Cloud console and Google Cloud CLI (Command Line Interface), will pay their hosting costs. In this analysis, different data files which are taken from publicly available cloud datasets are used.

By uploading files to a private server, an index table can be created. Multiple files are used for the System Analysis. The Info Documents are different sizes changing from 1 KB to 100MB. By encrypting data, this method protects the privacy of users more effectively and delivers higher-quality results. The upload time, encryption time, and recovery time of the files, throughput, scalability and availability are used to evaluate the effectiveness of the presented method.

File Upload time: The time taken by the cloud application to upload a user file on database.

Encryption Time: The time taken to encrypt the data files of user in cloud.

File recovering time: The time taken to recover the corrupted file.

Throughput: This measures the amount of data that can be processed by the system per unit time. Higher throughput is generally better.

Availability: This measures the percentage of time that the cloud service is available to users. Higher availability is generally better.

Scalability: scalability measures the ability of the system to handle an increasing number of users or data storage needs. A system with good scalability can accommodate growth without compromising performance.

The Fig. 2 shows the upload time for different sizes of files.

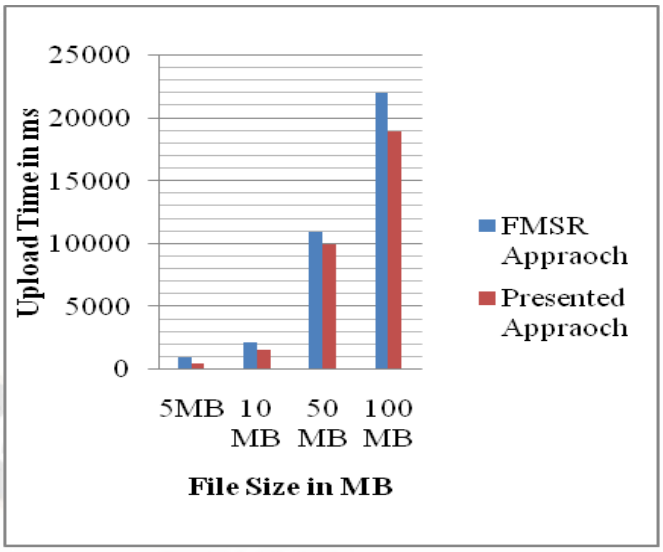

Fig. 2: Comparative Graph for Upload Time

Compared to Functional Minimum Storage Regeneration (FMSR) approach, presented approach requires less time for file uploading. The encryption and decryption times for various file size are shown in Fig. 3. In fig. 3, the X-axis represents different file sizes such as 5MB, 10MB, 50 MB and 100 MB whereas y-axis represents time in milli seconds. The size of the file has an impact on whether it requires time to encrypt and decrypt data.

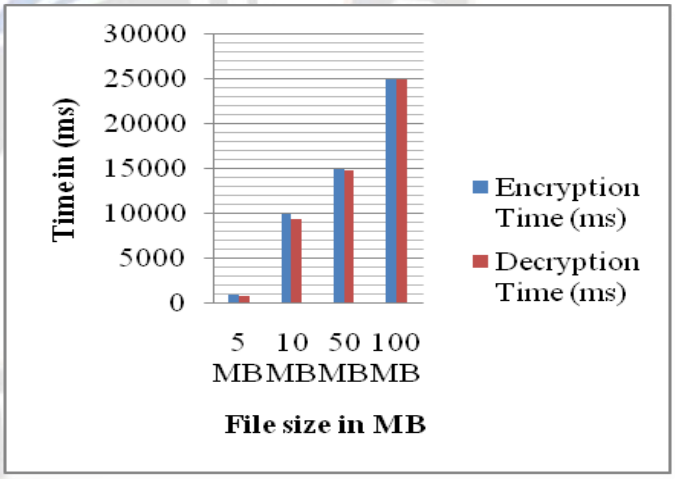

Fig. 3: Encryption and Decryption Time Comparison

Compared to encryption time, decryption time is less. The fig. 4 shows the recovering time of presented approach.

***Article Received:** 26 June 2023 **Revised:** 24 August 2023 **Accepted:** 04 September 2023*

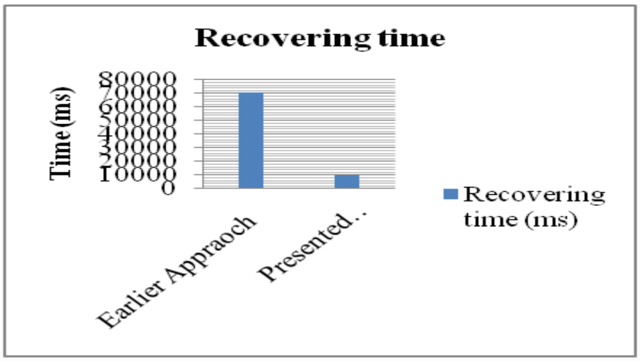

Fig. 4: Recovering time Comparison

In Figure 4, the x-axis shows several cloud storage strategies, and the y-axis shows recovery time in milliseconds. Compared to earlier approach i.e. Public Auditing for Regenerating Code Based Cloud Storage approach, presented approach recovers the corrupted file very accurately within less time. Even some of the State-of-art approaches didn't get the corrupted file. However presented Implementation of dynamic virtual cloud environment for privacy data storage approach has effectively recovered the corrupted files. The Fig. 5 shows the comparative graph for scalability and throughput.

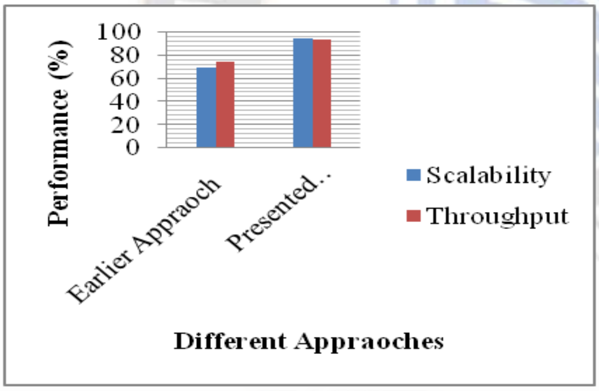

Fig. 5: Scalability and Throughput Comparison

Presented virtual cloud storage approach has provided high throughput and scalability than earlier approach. The Fig. 6 shows the availability performance comparsion.

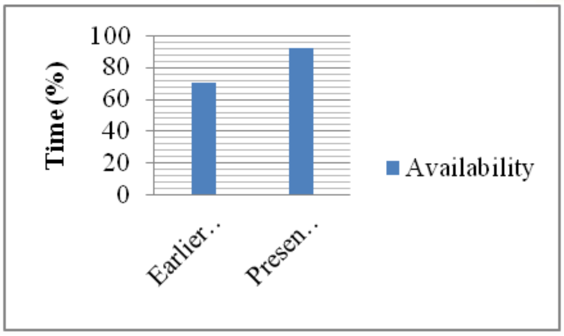

Fig. 6: Comparative graph for Availability

In fig. 6 the x-axis represents different approaches whereas y-axis represents the available time of cloud in terms of percentage. Compared to earlier approach, presented approach has high availability. The Fig. 7 shows the authenticity and confidentiality comparison of different cloud storage approaches. Presented virtual cloud storage approach has provided better confidentiality and authenticity to user data than earlier approach.

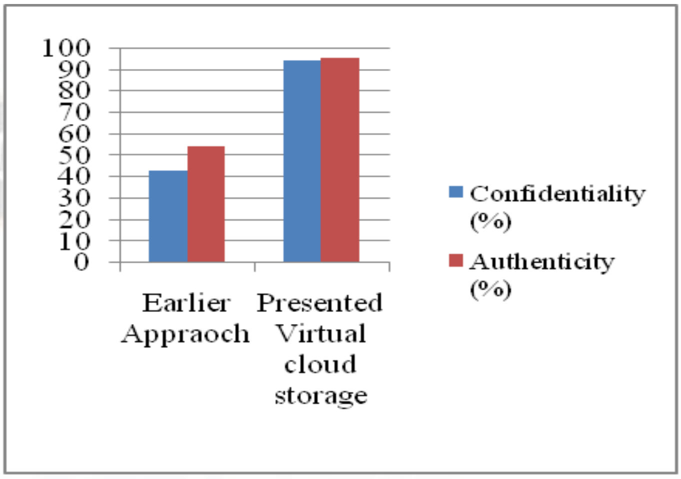

Fig. 7: Performance Comparison

Hence the implementation of dynamic virtual cloud environment for privacy data storage has achieved great authenticity, confidentiality and high scalability, throughput availability. In addition it takes less time for encryption, decryption as well as corrupted file recovering.

V. CONCLUSION

The implementation of dynamic virtual cloud environment for privacy data storage is presented in this work. This system will connect the local server to the cloud server so that users can share data. Some authentication is required for particular data or information. Encryption is used to process this authentication. Before uploading to the cloud, user files are encrypted, and an encrypted index tree is created for the security of user data. The file's size ranges from 1 kilobyte to 100 megabytes. To maintain integrity, a public audit method based on a Third-Party A slightly confident auditor is selected. A regenerative-code-based proxy server that reconstructs the original data from its hash values are included with this TPA. When any blocks are damaged or lost, this system uses a replacement coding technique at the proxy to retrieve the corrupted data blocks. The encryption time, decryption time, corrupted file recovery time, availability, scalability and throughput are all used to evaluate the performance of the presented method. Compared to state-of-art approaches, presented Implementation of dynamic virtual cloud environment for privacy data storage approach requires less time for uploading, encryption, decryption and file recovering. In addition, this system has better confidentiality and authenticity than state-of-art approaches. This dynamic virtual cloud environment will be a better solution for securing the privacy of user data in cloud storage environments. In future Privacy-preserving in cloud computing for data storage security framework using regenerating homomorphic encryption will be presented which will provide better privacy preservation for cloud storage systems.

REFERENCES

[1] M. Antony Joans Kumar, C. Christopher Columbus, E. Ben George and T. Ajith Bosco Raj, "A Virtual Cloud Storage Architecture for Enhanced Data Security", Computer Systems Science & Engineering, CSSE, 2023, vol.44, no.2, DOI: 10.32604/csse.2023.025906

[2] Banoth Anantharam, Dr.Neeraj Sharma, Dr. B.Kavitha Rani, " Design A Privacy-Preservation Approach used in Public Auditing for Regenerating-Code-Based Cloud Storage", International Journal of Mechanical Engineering, Vol. 6 No. 3 December, 2021, ISSN: 0974-5823

[3] Guangjun Liu, Wangmei Guo, Ximeng Liu, and Jinbo Xiong, "Security Analysis and Improvements on a Remote Integrity Checking Scheme for Regenerating-Coding-Based Distributed Storage", Hindawi Security and Communication Networks, Volume 2021, Article ID 6652606, 8 pages, doi:10.1155/2021/6652606

[4] Dr.Neeraj Sharma, Dr. B.Kavitha Rani, "Advanced Algorithm For Privacy Preservasing In Regenerating Code Based Cloud Storage", Journal of Critical Reviews Issn- 2394-5125 VOL 7, ISSUE 19, 2020

[5] Jindan Zhang, Rongxing Lu, Baocang Wang, Xu An Wang, "Comments on "Privacy-Preserving Public Auditing Protocol for Regenerating-Code-Based Cloud Storage", IEEE Transactions on Information Forensics and Security (Volume: 16), **DOI:** 10.1109/TIFS.2020.3032283

[6] Geeta C M, Tejashwinishivaram K S, Shreyas Raju R G, Raghavendra S, Rajkumar Buyyay, Venugopal K Rz S S Iyengarx L M Patnaik, "SRCBT: Secure Regeneration of Corrupted Blocks by TPA in Cloud", 2020 IEEE Region 10 Symposium (TENSYMP), 5-7 June 2020, Dhaka, Bangladesh, 978-1-7281-7366-5/20

[7] Wei Liang, Yongkai Fan, Kuan-Ching Li, Dafang Zhang, and Jean-Luc Gaudiot, "Secure Data Storage and Recovery in Industrial Blockchain Network Environments", IEEE Transactions on Industrial Informatics, Vol.16, No.10, 2020, **DOI:** 10.1109/TII.2020.2966069

[8] R.Naveenkumar and K.Muthusamy, N.Arun Prasath and R.Krishnaraj "Deduplication And Security Enhancement In Cloud Computing", International Journal o f Electrical Engineering and Technology (IJEET), Volume 10, Issue 6, November-December 2019, pp.5 4-60,

[9] Jyoti Mahajan, "Public Auditing for Regenerating Code Based Cloud Storage", International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 04 Issue: 07 | July -2017

[10] Miss. Nirupamashree, Mrs. Pushpa R, "A Secured Public Auditing for Regenerating-Code-Based Cloud Storage", IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 3 Issue 6, June 2016, ISSN (Online) 2348 – 7968

[11] Aisha Ahmed, Machine Learning in Agriculture: Crop Yield Prediction and Disease Detection , Machine Learning Applications Conference Proceedings, Vol 2 2022.

[12] Aaziz Fadhil , R. ., & Haddi Hassan, Z. A. . (2023). A Hybrid Honey-Badger Intelligence Algorithm with Nelder-Mead Method and Its Application for Reliability Optimization. International Journal of Intelligent Systems and Applications in Engineering, 11(4s), 136–145. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/2580

[13] Rajesh M. Patil, Prof. Ismail Mohamed, "Privacy-Preserving Public Auditing for Regenerating-Code-Based Cloud Storage", International Journal of Innovative Research in Science, Engineering and Technology, Vol. 5, Issue 6, June 2016

[14] Pritha, K. and Nivethitha, M, "Secure Sharing in Cloud Using Regenerated Code", International Journal of Current Research, Vol. 8, Issue, 04, pp.29487-29489, April, 2016, ISSN: 0975-833X

[15] Shenling Liu, Chunyuan Zhang, Le Bo, "Improve Security And Availability For Cloud Storage", Proceedings of CCIS2016, 978-1-5090-1256-5/16, 2016 IEEE

[16] P. Balasermathi and Mrs.S.G. Sandhya, "Security Analysis of Data Integrity In Regenerating Coding Based Cloud Storage", International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE) ISSN: 0976-1353 Volume 21 Issue 3 – APRIL 2016,

[17] Dr.V.Goutham, J.Rachana, M.Nikhila, "Framework for Secure and Dynamic Auditing for Regenerating Code Based Data Storage in Cloud Platform", International Journal of Scientific & Engineering Research, Volume 7, Issue 8, August-2016, ISSN 2229-5518

[18] Jing Chen, Yuling Peng, Ruiying Du, Quan Yuan and Minghui Zheng, "Regenerating-Codes-based Efficient Remote Data Checking and Repairing in Cloud Storage", 2015 IEEE Trustcom/BigDataSE/ISPA, 978-1-4673-7952-6/15, 2015 IEEE DOI 10.1109/Trustcom-BigDataSe

[19] Kamini Suresh Bhavsar, "Regenerating-Code-Based Cloud Storage using Privacy-Preserving Public Auditing", International Journal on Recent and Innovation Trends in Computing and Communication, 2015, ISSN: 2321-8169 Volume: 3 Issue: 12

[20] Mr. Kunal Verma, Mr. Dharmesh Dhabliya. (2015). Design of Hand Motion Assist Robot for Rehabilitation Physiotherapy. International Journal of New Practices in Management and Engineering, 4(04), 07 - 11. Retrieved from http://ijnpme.org/index.php/IJNPME/article/view/40

[21] Dongju Yang, Chuan Ren, "VCSS: An Integration Framework for Open Cloud Storage Services", 2014 IEEE 10th World Congress on Services, 978-1-4799-5069-0/14, 2014 IEEE, DOI:10.1109/SERVICES.2014.36

[22] Henry C.H. Chen, Yuchong Hu, Patrick P.C. Lee, and Yang Tang, “NCCloud: A Network-Coding-Based Storage System in a Cloud-of-Clouds”, IEEE Transactions On Computers, Vol. 63, NO. 1, JANUARY 2014

[23] Henry C. H. Chen and Patrick P. C. Lee, “Enabling Data Integrity Protection in Regenerating-Coding-Based Cloud Storage”, 2012 31st International Symposium on Reliable Distributed Systems, 1060-9857/12, 2012 IEEE, DOI 10.1109/SRDS.2012.24