SJIF Impact Factor 2022: 8.197| ISI I.F. Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 7 | Issue: 4 | April 2022 - Peer Reviewed Journal

DATA DEDUPLICATION ON FILE SYSTEM USING FUSE

Shashikant Yadav[1], Mairaj Ali[2], Neeraj Gupta[3], Smruti Patil[4]

[1]*Student, Dept. of I.T. VPPCOE & VA, Mumbai UniversityMumbai, India*
[2]*Student, Dept. of I.T.VPPCOE & VA, Mumbai UniversityMumbai, India*
[3]*Student, Dept. of I.T.VPPCOE & VA, Mumbai University Mumbai, India*
[4]*Professor, Dept. of I.T VPPCOE &VA,Mumbai University Mumbai, India*

ABSTRACT

For any organization or any of business community their data is must needed thing to take next decision or forward strategies . As just now days in market many of platforms are available which works on previous data or using previous data they can forward but there would be problem about large amount data where they can store then there would be many resource like cloud computing or etc. but using thing they have paid for these things how much they can use it. For any organization there would have same data in repeating manner but can't clarify it because of it takes more time. Apart of that as example if we had to conduct any exam in online mode and we launch the software at each computer to conduct exam it alsotake more time and man power and it's very complex.to elaborate all these problem which thing mention we have data deduplication on file system using fuse by these can operate many file from one computer any by we can rectify the duplicate data and solve thehaving by these by this whole solution here we have good advantage like less time consumption, less cost , less man power need. There have many of advantages. This task is achieved with varying degree of success through the implementation of data deduplication on file system using fuse.

KEYWORDS - *Time complexity, Less storage, Fuse library, SHA-256 Algorithms, data analysis, Database, libraries, result.*

I. INTRODUCTION

Data Deduplication is complicated technologies that may dramatically scale back the quantity of backup knowledge hold on by eliminating redundant knowledge.

The deduplication method needs compression {of knowledge of knowledge of information} chunks which are a singular contiguous block of data; these chunks are known and hold on throughout the method of research and compared to different chunks among existing data.

Whenever a match happens the redundant chunk is replaced with a tiny low reference that points to the hold on chunks.

It is Technology which is able to for for sure amendment maps name to associate degree object and object to file contents. as an example, a file 'abc.txt' could be a file name that maps to a file that eventually contains file knowledge. Thus, classification system provides the way to prepare, store, retrieve and maintain info with the services like open(), read(), write(), close() among several.

all the Tradition methodology to store and retrieve knowledge, as in forthcoming days storing would possibly become an enormous downside which might be simply solved exploitation knowledge Deduplication methodology up to some extent, additionally can facilitate correct knowledge storing. It Uses FUSE for writing Virtual classification system. in contrast to ancient file systems that primarily work with knowledge on mass storage, virtual filesystems do not truly store knowledge themselves.

A Filesystem manages knowledge on a ADPS. Filesystem is organized as a ranked system with directories at the highest, containing a collection of files, and a file itself being a group of information blocks. Filesystem primarily

II.LITERATURE SURVEY

B. Thakkar and B. Than Kachan [1], created a comparison of cryptanalytic algorithms like AES, RSA, IDEA, Blowfish and DES over cloud during a survey. The comparison of cryptanalytic algorithms were supported the factors of coding type, key size, block size, variety of rounds used, execution

SJIF Impact Factor 2022: 8.197| ISI I.F. Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 7 | Issue: 4 | April 2022 - Peer Reviewed Journal

time, memory usage and encodingcapability. They concluded in their survey that interchangeable algorithms ar a lot of economical. On the idea on memory usage Blowfish was best and supported coding time, AES and Blowfish were quick.

V.Radia and D. Dingh[2] made a study of data deduplication techniques: file level, block level, inline post process, source based and target based. The study concluded that source-based deduplicationis best to optimize upload bandwidth and storage space over cloud. Distributed deduplication provides security and confidentiality. Both approaches together provide reliability.

N. Pachpor and P. Prasad[3], projected a newPerformance-Oriented knowledge(POD)deduplication technique and conjointly made comparative analysis with different existing techniques. The new theme removes duplicate files and duplicate knowledge

L. Suresh and M. Bharathi [4], made a study of various data deduplication techniques and issues of data deduplication on cloud storage. A new strategy was proposed that allows for fast data transfer from clientto server, use of new hash algorithm and removal of duplicate data using block level deduplication.

K. Akhila et al.[5**],** study of various data deduplication techniques like ClouDedup, DupLESS, HEDup, andSecDup was done. Algorithms were based on convergent key method. Authors stated that a good strategy for enhanced storage optimization technique could be used.

W. Kim and I. Lee[6], have discussed various ways of data deduplications sites and level. Issues regarding secure data deduplications like data encryption, dictionary attacks and poison attacks were put forward. The authors also discussed security techniques for achieving secure deduplication like use of convergent encryption and currently used secure deduplication systems like DupLESS, ClouDedup and PerfectDedup.

A.Nair et al.[7], proposed three protocols for the process namely File uploading protocol, Integrity auditing protocol and Proof of ownership protocol. In first protocol, client generates hash for chunks of a file using SHA-1 algorithm, checks if hash already present on cloud and if found third protocol would be called. If no hash is found on cloud, client sends chunks to auditor, where auditor creates tags for chunks, encrypts the chunks using AES algorithm and compresses it using Deflate algorithm that is combination of Huffman coding and LZ77 and later sends the tags and chunks to cloud. Second protocol is used for verifying integrity, which is done by the cloud server. Client or auditor asks for verification or establishing proof and server verifies the same. Third protocol is used to identify the ownership of the file where server verifies the client. The authors applied the scheme on file of various sizes and achieved utilization of efficient space

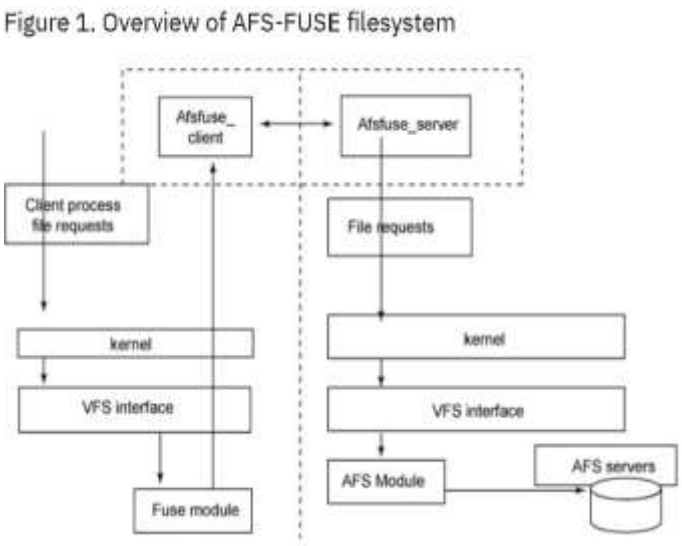
J. Malhotra and J. Bakal[8], identified various challenges of deduplication, comparison of current deduplication techniques were made based on chunking method, metadata processing and throughput. Two Threshold Two Divisor algorithm with Switch Divisor and Two Threshold Two Divisor algorithm were used on different file sizes, analysis was made on time taken for deduplication, and it was identified that Two Threshold Two Divisor algorithm with Switch Divisor takes less time. The authors proposed that throughput could be achieved by using parallelized deduplication process.

S. Sathe and N. Dongre[9], applied block level data deduplication strategy. The user registration is done before uploading of the file. The file is then fragmented into fixed size blocks. Before uploading the file, the hash value is generated using SHA-512 algorithm and this hash value is compared with already existing file fragments on cloud. If no match is found or the count of duplicate block is below a predefined threshold value, the file is uploaded on server. After the file is uploaded, it is encrypted using AES algorithm and stored. The file can be downloaded later if proper key attributes are provided. The server can also delete the file only after the owner is verified by using the policy based file assured deletion method.

III. PROPOSED SYSTEM

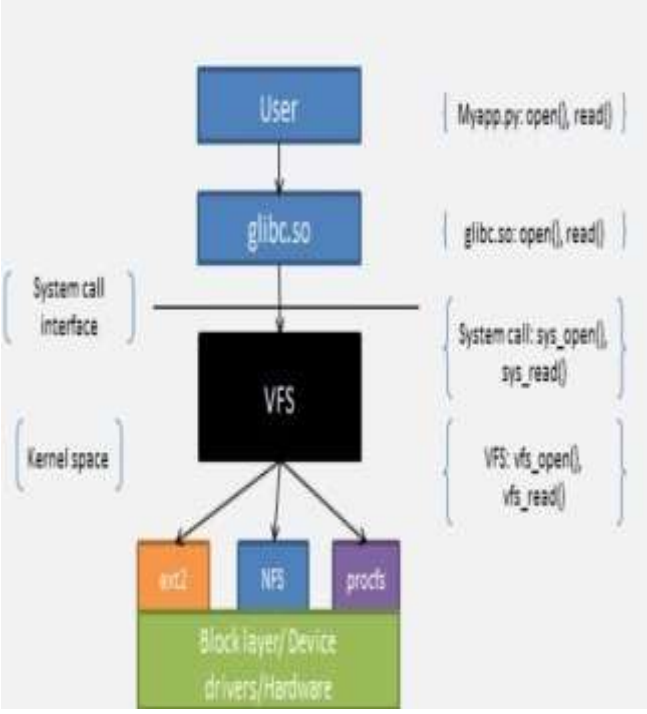
we have make a filesystem in fuse using python and implemented all method for filesystem and also implemented a block level data deduplication using **SHA-256** and used a back-end **MYSQL** to take backup of file Software used Tech: Python, FUSE, MYSQL SHA-256,MYSQL-Connector Hardware used Processor: i3 4th Generation or higher, 4GB RAM, 1GB Storage.

SJIF Impact Factor 2022: 8.197| ISI I.F. Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 7 | Issue: 4 | April 2022 - Peer Reviewed Journal



File system in UNIX: Virtual Filesystem

In OS kernel, a Filesystem implementation is abstracted with a virtual Filesystem (VFS) . VFS is Associate in Nursing umbrella that acts as Associate in Nursing interface to all or any on the market (mounted) Filesystems on a system. Its major task is to: Decouple Filesystem operation from the interface - Manage Filesystem 'mount' - puzzle out the target Filesystem for Associate in Nursing file I/O request and route the request - give a uniform interface (POSIX) to user application for file I/O Filesystems on *NIX system get registered with VFS with a mount purpose. What happens once a user application tries to access a file residing on (e.g. NFS partition) Filesystem? Let's .



SJIF Impact Factor 2022: 8.197| ISI I.F. Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 7 | Issue: 4 | April 2022 - Peer Reviewed Journal

Converting the kind of a folder system

It may be necessary to possess folders during a completely different filing system that they presently exist reasons embrace to requirement for a rise within the house necessities on the far side the boundaries of the present filing system the depth of route may have to be exaggerated on the far side of restrictions to thefiling system there is also performances or responsible concerns providing access to a different operating system that doesn't support the present filing system is one more reason

Migrating to a unique filing system

Migrate the disadvantage of requiring further house though it's going to be quicker the simplest case is that if there unused house on media which can contain the ultimate filing system for example to migrate a fat32 filing system They are often filled and mount at normal user thus their suitable for folder system that users favor mount by themselves for network entrance for researching archive files for demountable media etc If a fuse file system pilot collision it will not read us kernel you will notice nothing worse than io errors at application where accessing the file system Their often programme terribly rapidly there square measure fuse binding to several writing language wherever a helpful fuse file system pilot are often written in an exceedingly few hundred lines of code initial produce a brand new ext2 filing system then copy the info to the filing system Then delete the fat32 filing system different once there insufficient house to recover the first filing system till the new one is made is to use a piece space such as a removable media this takes longer however a backup of the info could be a nice aspect impact In-place conversion in some cases conversion is done in-place though migrating the filing system is a lot of conservative because it involves a making a duplicate of the info and is suggested on windows fat and fat32 file systems is regenerate to ntfs via the convert exe utility however not the reverse on Linux ext2 is regenerate to ext3 and regenerate back and ext3 is regenerate to ext4 but not back and each ext3 and ext4 is regenerate to and regenerate back till the undo info is deleted these conversions area unit doable because of victimization identical format for the file information itself and relocating the data into empty house in some cases victimization support. We implemented Data deduplication by using these two types of method

File Level Data Deduplication

As the title it totally works on only files. In this method comparing the file with existing file and after that if their match occur with existing files then it eliminate it otherwise it store the file on cloud File Level Data Deduplication only works on files not works on data so that it can't eliminate all duplicate value

Block Level Data Deduplication

As it name totally works on blocks and the blocks "files divided into small chunks is called blocks" here blocks compare with existing blocks over cloud if the match occur then it store otherwise remove the similar data.
This technique takes more time but it completely remove the duplicate value.

The implementation follows these steps
Step 1: we install fuse python:pip install fuse
Step 2: we have install mysql connector: pip install mysql connector
Step 3: database setup
We CREATE DATABASE mydb;
CREATE TABLE mytable(key VARCHAR(64) PRIMARY KEY, blocks CHAR(4096));

IV RESULT

Unix file system square measure historically enforced within to kernel fuse permits filesystems to be enforced by user program
In kernel folder systems square measure higher fitted to leading file systems for a plan and data

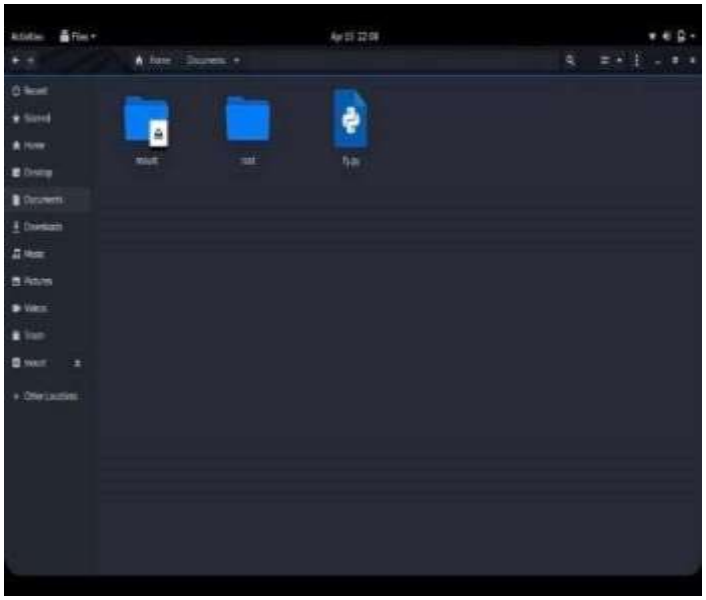
There are often used on field boot method that plan implement a fuse file system should be loaded from somewhere.

They are additional sturdy in this it will not get far because of a method blooming or being killed by error.

They are somewhat quicker fuse filesystems produce extra blessings principally revolves around their pliability

They are often deploy terribly rapidly each as a result of there's no want for administrate os interfaces to put in it and since they'll be moved simply between os There aren't any permit problems associated with being fixed connected with kernel has effects on

SJIF Impact Factor 2022: 8.197| ISI I.F. Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 7 | Issue: 4 | April 2022 - Peer Reviewed Journal



SJIF Impact Factor 2022: 8.197| ISI I.F. Value:1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 7 | Issue: 4 | April 2022 - Peer Reviewed Journal

V. CONCLUSION

We have described the operation, performance, and convenience of a transparent, adaptive mechanism for file system discovery and replacement. The adaptiveness of the method lies in the fact that a file service client no longer depends solely on a static description of where to find various file systems, but instead can invoke a resource location protocol to inspect the local area for file systems to replace the ones it already has 1.The observation that file system switching might be needed and useful. 2.The idea of an automatically self-reconfiguring file service, and of basing the reconfiguration on measured performance. 3.Quantification of the heuristics for triggering a search for a replacement file system. 4.The realization that a ``hot replacement'' mechanism should not be difficult to implement in an NFS/ vnodes setting, and the implementation of such a mechanism

REFERENCES

1. *B. Thakkar and B. Thankachan, "A Survey for Comparative Analysis of various Cryptographic Algorithms used to Secure Data on Cloud," Int. J. Eng. Res. Technol., vol. V9, no. 08, pp. 753–756, 2020, doi: 10.17577/ijertv9is080328*
2. *V. S. R. and D. K. Singh, "Secure Deduplication Techniques: A Study," Int. J. Comput. Appl., vol. 137, no. 8, pp. 41–43, 2016, doi: 10.5120/ijca2016908874.*
3. *N. N. Pachpor and P. S. Prasad, "Securing the Data Deduplication to Improve the Performance of Systems in the Cloud Infrastructure," in Performance Management of Integrated Systems and its Applications in Software Engineering, Springer Singapore, 2020, pp. 43–58*
4. *L. S. and M. A. Bharathi, "Analysis of Block- Level Data Deduplication on Cloud Storage," Ambient Commun. Comput. Syst., vol. 904, no. July, pp. 401–409, 2019, doi: 10.1007/978-981-13-5934-7*
5. *K. Akhila, A. Ganesh, and C. Sunitha, "A Study on Deduplication Techniques over Encrypted Data," in Procedia Computer Science, 2016, vol. 87, pp. 38–43, Doi: 10.1016/j.procs.2016.05.123.*
6. *W. Bin Kim and I. Y. Lee, "Overview of Data DeduplicationTechnology in a Cloud Storage Environment," in Lecture Notes in Electrical Engineering, 2020, vol. 536 LNEE pp. 465–470, Doi: 10.1007/978-981-13-9341-9_80.*
7. *R. P. J. and P. S. L. K. Arya S. Nair, B. Radhakrishnan, "Secure Data Deduplication and Efficient Storage Utilization in Cloud Servers Using Encryption, Compression and Integrity Auditing," Int. Conf. Soft Comput. Syst., vol. 837, pp. 326–334, 2018, Doi: 10.1007/978-981-13-1936-5.*
8. *J. Malhotra and J. Bakal, "A survey and comparative study of data deduplication techniques," 2015 Int. Conf. Pervasive Comput. Adv. Commun. Technol. Appl. Soc. ICPC 2015, vol. 00, no. c, pp. 0–4, 2015, Doi: 10.1109/PERVASIVE.2015.7087\and assured deletion in cloud," in Proceedings of the International Conference on Smart Systems and Inventive Technology, ICSSIT 2018, 2018, no. Icssit, pp. 406–409,Doi: 10.1109/ICSSIT.2018.8748482.*