Improving the Reliability of Data Deduplication Based on Storage

P.Divya[1], R.Kowsalya[2], S.A. Sara Irsath Nowsin[3], Ms.T.Suvaikin Punita[4], Mrs.Dr.P.Gomathi[5]
[1, 2, 3] Dept of Computer Science and Engineering
[4, 5] Faculty, Dept of Computer Science and Engineering
[1, 2, 3, 4, 5] N.S.N. College of Engineering and Technology, Karur, India

Abstract- *Data deduplication weakens the reliability of storage systems since by design it removes duplicate data chunks common to different files and forces these files to share a single physical date chunk, or critical chunk, after deduplication. Thus, the loss of a single such critical data chunk can potentially render all referencing files unavailable. Existing approaches introduce data redundancy after files have been deduplicated, either by replication on critical data chunks, i.e., chunks with high reference count, or RAID schemes on unique data chunks, which mean that these schemes are based on individual unique data chunks rather than individual files. Per-file parity scheme to improve the reliability of deduplication. PFP computes the XOR parity within parity groups of data chunks of each file after the chunking process but before the data chunks are deduplicated. Therefore, PFP can provide parity redundancy protection for all files by intra-file recovery and a higher-level protection for data chunks with high reference counts by inter-file recovery. It tolerates the multiple data chunk failures and guaranteeing file availability upon multiple data chunk failures. The deduplication system reduce the 75% of replica compare with various techniques and also give the ensure reliability for stored data.*

Keywords- Data deduplication, Data chunk, Inter file recovery, Intra file recovery, Per-file parity, Reliability

I. INTRODUCTION

It is the most critical challenges for the design and management of large-scale storage systems in face of the explosive growth in data volume. Consequently, data reduction technologies have been propelled to the forefront of research and development in addressing this challenge in the big data era. Data deduplication, a space efficient data reduction technology, has spurred a great deal of research interest from both industry and academia. It has been deployed in a wide range of storage systems, including backup and archiving systems and primary storage systems such as VM (Virtual Machine) servers.

The existing system on data deduplication focus on improving its efficiency by developing or optimizing chunking schemes to find as much redundant data as possible, solving the index- lookup disk-bottleneck problem, and addressing the hash computing over-head issue and the data restore problem. However, the impact of data deduplication on the reliability of the stored data has not been well understood nor studied for large-scale storage systems. This because reliability is often synonymous with redundancy and data deduplication, data redundancy is completely eliminated by its very design. In other words, since only a single copy of duplicate data common to and referenced by different files, also referred to as a critical data chunk, is stored in the persistent storage after deduplication. The loss of one or a few critical data chunks can lead to many referencing files to be lost, thus significantly reducing the reliability of the storage system. Moreover, in a large-scale storage system, a post-deduplication file may have its constituent data chunks stored on multiple different storage devices. If any one of these constituent data chunks or storage devices fails, the file is lost. Therefore, data deduplication magnifies the negative impact of data loss in large-scale storage systems.

1.1 Cloud computing

Cloud computing is basically a service oriented architecture rendering easy access to all who make use of it. The need of computation power rendered by the machines is on a Continuous hike nowadays. The CPU computation power is boosting twice for every 3 years. However size of the files keeps increasing also in an amazing rate. 20 years ago, the common format is only text file. Later, computer can handle the graphics well, and play some low quality movies. In recent year, human is not satisfied in the quality of DVD, and introduce the Blue- ray disk. The file is changed form a few KBs to now nearly 1 TB. The varying characteristics of cloud making it different from other computing technologies are on-demand self-service, agility, autonomic computing, virtualization, parallel and distributed architecture and pay for use model.

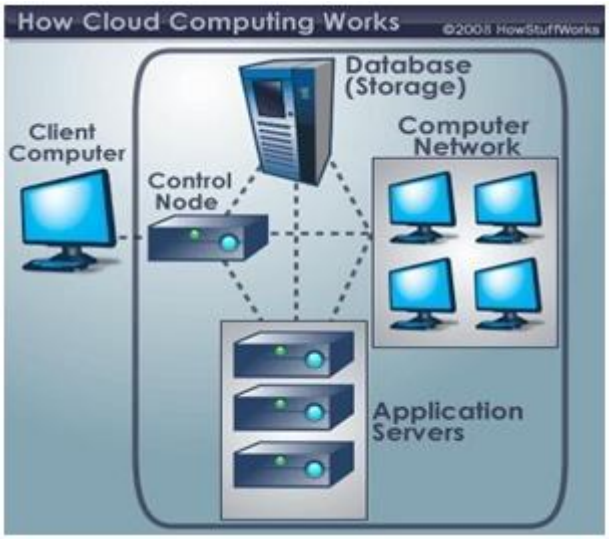

Fig: 1. Working of cloud computing

1.2 Types of cloud services

Infrastructure as a service (IaaS)

- The customer can access the providing requests, data storage, networks, and other essential computing resources.
- Consumer is able to organize and sprint random software, along with the Operating System and application.
- The cloud infrastructure is not managed and controlled by the consumer but they can access operating system, storage, organized applications, and probably incomplete control of select networking machinery like host firewalls.

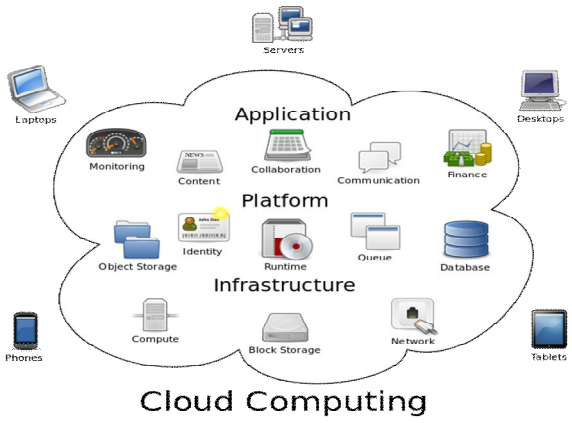

Fig: 2. Types of cloud services

Platform as a service (PaaS)

- The qualifications afforded to the end user is to arrange onto the cloud communications purchaser created or acquired request created using programming expression and mechanisms reinforced by the contributor.
- The end user does not manage or control the fundamental cloud communications including network, servers, operating systems, or storage space, but has control over the arranged applications and possibly product hosting location patterns.

Software as a service (SaaS)

- The facility presented to the end user is to utilize the contributor's products running on the cloud transportation.
- The appliances are available from diverse punters approach through a thin client crossing point such as a web browser i.e., web-based email.
- The end user does not manage or control the essential cloud transportation including system, servers, operating systems, storage space, or even human being function capacity, with the potential omission of restricted user-specific function relationship.

II. EXISTING SYSTEM

Existing approaches to address the reliability problem in deduplication-based storage systems can be classified into two categories, namely, deduplication-then-RAID (DTR), where the stored unique data chunks are organized and thus protected by a RAID scheme, and reference-count based replication (RCR), where data chunks with sufficiently high reference counts (i.e., number of different files sharing/referencing the same data chunk) are replicated on different storage devices. While approaches in both categories introduce data redundancy for reliability of deduplication-based storage systems, they do so after files have been deduplicated and generate data redundancy based on the stored unique data chunks to prevent loss of individual data chunks rather than individual files. In other words, files can still be vulnerable to data loss because of the specific ways in which a data chunk is protected and a storage device fails. There are generally two types of storage device failures, namely, disk failures necessitating a disk replacement where all stored data on the failed disk are considered lost, and errors in individual data blocks that cannot be recovered with a re-read or the sector-based error correction code (ECC), errors often referred to as latent sector errors. While the DTR approaches provide RAID protection against one or two disk failures, they can suffer from data loss in face of multiple concurrent latent sector errors or unrecoverable read errors within a stripe. On the other hand, the RCR approaches, while providing good protection for data chunks with high reference count, can suffer from file unavailability if any of a file's

constituent data chunks with low reference count are lost due to device failures, latent sector errors or unrecoverable read errors.

These data block failures, rather than drive failures, suggest that providing protection against only disk failure is not sufficient to prevent file losses. This is because upon a disk failure, an unrecoverable block read error on any of the active disks during RAID reconstruction would lead to data loss. The same problem occurs when two disks fail under a RAID6 scheme. Similarly, only protecting data chunks with high reference counts is insufficient to prevent individual files from becoming unavailable, as discussed above. In contrast, protecting a file in its entirety before it is deduplicated is arguably much more effective in avoiding both file and data chunk failures. The reason is that by applying redundancy protection within a file, a certain number of data chunk failures can be recovered within a file depending on the redundancy scheme and a critical data chunk with a high reference count can be covered by any of the multiple parity groups belonging to as many files as the chunk's reference count.

2.1 Disadvantages

- Network congestion is high and creation time is more.
- It affects overall data availability due to slow and long process.
- Fails to improve fault tolerance and reduces end user latency.

III. PROPOSED SYSTEM

It propose per- file parity (PFP) to improve the reliability of deduplication-based storage systems. PFP computes the parity for every N chunks, where N is a configurable parameter, or for a whole file. When a disk failure is detected, the generated parity chunks can be used to recover from the read errors and failed data chunks by intra-file recovery. On the other hand, when several errors occur in a parity group of parity and these failed data chunks each have reference counts of greater than 1, PFP can recover these failed data chunks by inter-file recovery by leveraging the parity groups of the unaffected referencing files.

The reliability results show that PFP can tolerate much more data chunk failures and guarantee file availability upon multiple data chunk failures. Moreover, the failure-injection based evaluations show that the PFP scheme can tolerate hundreds of concurrent chunk errors without file loss for data sets with high deduplication ratios. The evaluations also show that PFP is highly cost effective in terms of file-loss-tolerance/redundancy measure by an average of 52.2% and 197.5% over the DTR and RCR schemes, respectively. On the other hand, the performance assessment shows that PFP's significant reliability gain comes at an acceptable performance cost of an average of 5.7% performance degradation to the deduplication-based storage system.

The data stored in the cloud can be retrieved and the integrity of these data can be ensured. It was based on pseudo random function and BLS signature, a private remote data integrity auditing scheme and a public remote data integrity auditing scheme. To protect the data privacy, a privacy-preserving remote data integrity auditing scheme with a random masking technique has been used. To reduce the burden of signature generation we designed a remote data integrity auditing scheme based on the in distinguish ability obfuscation technique. A third party medium (TPM) is designed a light-weight remote data integrity auditing scheme. In this scheme, the TPM helps user generate signatures on the condition that data privacy can be protected. The data sharing is an important application to protect the identity privacy of user. At the same time data has been saved in to fog server as a temporary data which reduces the risks of data loss or data damage in the cloud server. By using the fog and cloud server's data can be recovered from cloud or the fog server in an efficient way.

3.1 Advantages

- Avoids the redundancy when copying replicas to decrease bandwidth consumption
- Carefully select the source replicas in order to reduce the migration time
- Ensure the replica availability during the data migrations
- Avoids idle times in order to reduce the overall migration time
- Reduces the damage and loss of data
- Increases the efficiency
- Data are maintained with appropriate security

IV. SYSTEMDESIGN

4.1 System architecture

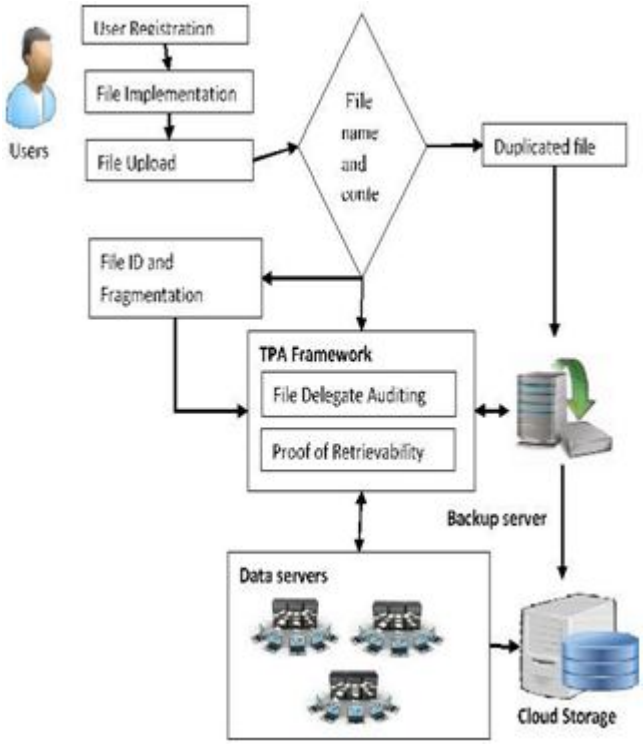

Fig: 3. Architecture of data deduplication

The system architecture will explain the overall process of data deduplication. They registered user to store the data in cloud. Then updated data checked by this method. After that checking process, if the duplicated file not stored in the cloud storage. The third party auditor will verify the already stored data from the cloud storage.

V. METHODS USED

5.1 Advanced Encryption Standard (AES)

The AES algorithm is used to encrypt the stored data.

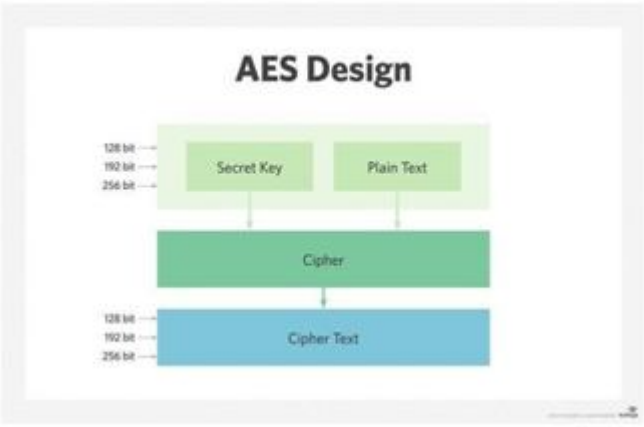

Fig: 4. Block diagram of AES

The AES encryption algorithm defines a number of transformations that are to be performed on data stored in an array. The first step of the cipher is to put the data into an array; after which, the cipher transformations are repeated over a number of encryption rounds. The number of rounds is determined by the key length, with 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. The first transformation in the AES encryption cipher is substitution of data using a substitution table; the second transformation shifts data rows, the third mixes columns. The last transformation is a simple exclusive or operation performed on each column using a different part of the encryption key. Longer keys need more rounds to complete.

5.2 Third Party Auditing (TPA)

The third party auditor will verify the stored data after the auditing request.

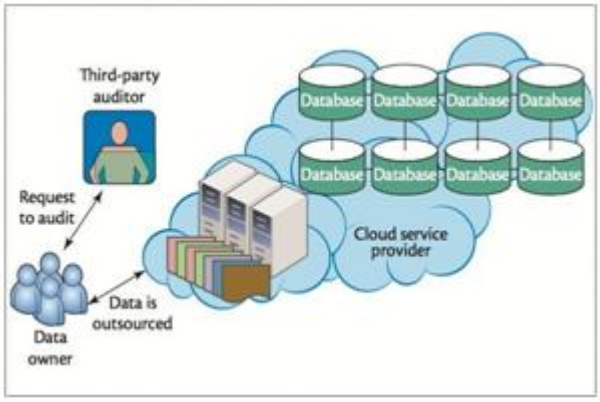

Fig: 5. Block diagram of TPA

The third-party auditor (TPA) to audit the outsourced data when needed**.** In this module, allow TPA to verify the correctness of the cloud data on demand without retrieving a copy of the whole data or introducing additional online burden to the cloud users. Through the organization of privacy-preserving public auditing in Cloud Computing, TPA may concurrently handle multiple auditing delegations upon different user requests. The individual auditing of these tasks for TPA can be and very difficult and inefficient. Batch auditing not only allows TPA to perform the multiple auditing tasks at the same time, but also greatly reduces the computation cost on the TPA side

VI. CONCLUSION

Data deduplication has been widely used to improve the storage efficiency in modern primary and secondary storage systems. While increasingly important, the reliability issue of deduplication-based storage systems has not received

sufficient attention. A per-file parity (PFP) scheme is proposed to improve the reliability of deduplication-based storage systems. PFP computes the parity for each parity group of *N* chunks (*N*−1 data chunks and 1 parity chunk, where N is configurable) within each file before the file is deduplicated. Therefore, PFP can provide redundancy protection for all files by intra-file recovery as well as a higher level of protection for data chunks with high reference counts, critical data chunks, by inter-file recovery. Along with that we proposed an identity-based data integrity auditing scheme for secure cloud storage, which supports data sharing with sensitive information hiding. In our scheme, the file stored in the cloud can be shared and used by others on the condition that the sensitive information of the file is protected. Besides, the remote data integrity auditing is still able to be efficiently executed. The security proof and the experimental analysis demonstrate that the proposed scheme achieves desirable security and efficiency

REFERENCES

[1] P. Shilane, R. Chitloor, and U. Jonnala, "99 Deduplication Problems," in HotStorage'16, Jun.2016

[2] W.Xia, H.Jiang, D.Feng, F.Douglis, P.Shilane, Y.Hua,M. Fu, Y. Zhang, and Y. Zhou, "A Comprehensive Study of the Past, Present, and Future of Data Deduplication," Proceedings of the IEEE, vol. 104, no. 9, pp. 1681–1710, 2016.

[3] J. Paulo and J. Pereira, "Efficient Deduplication in a Distributed Primary Storage Infrastructure," *ACM* Transactions on Storage, vol. 12, no. 4, pp. 1–35, 2016.

[4] H. Wu, S. Sakr, C. Wang, L. Zhu, Y. Fu, and K. Lu, "HPDedup: A Hybrid Prioritized Data Deduplication Mechanism for Primary Storage in the Cloud," in MSST'17, Jun. 2017.

[5] M. Fu, P. P. C. Lee, D. Feng, Z. Chen, and Y. Xiao, "A Simulation Analysis of Reliability in Primary Storage Deduplication," in IISWC'16, Sept.2016.

[6] B. Schroeder, R. Lagisetty, and A. Merchant, "Flash Reliability in Production: The Expected and the Unexpected," *in FAST'16, San Jose, CA, Feb. 2016, pp. 67–80.*

[7] X. Chu, I. F. Ilyas, and P. Koutris. Distributed Data Deduplication. Technical Report CS-2016-02, University of Waterloo, 2016.

[8] A comparative study on data deduplication techniques in cloud storage B.Tirapathi Reddy*1, U.Ramya2, Dr.M.V.P Chandra Sekhar3, Sep 2016

[9] X. Chu, I. F. Ilyas, and P. Koutris. Distributed Data Deduplication. Technical Report CS-2016-02, University of Waterloo, 2016.

[10] A comparative study on data deduplication techniques in cloud storage B.Tirapathi Reddy*1, U.Ramya2, Dr.M.V.P Chandra Sekhar3, Sep 2016