ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 5, Issue 6, June 2016

Data Recovery Using Bi-Methodology Erasure Code For Reconstruction System In Cloud

A. Aruna Devi [1], D. Durai Kumar[2]

***Abstract*—Erasure coded storage cluster have increasingly become cost effective and fault tolerance solution for archive storage, data center, cloud storage. It protects against data loss caused by node failure because high reliability is an indispensable requirement for building large scale storage system. PULL - rep and PULL - sur are two existing reconstruction scheme based on PULL type transmission, where a rebuilding node initiates reconstruction by sending read request to fetch / pull surviving blocks such a pull node not only raises the TCP incast problem due to its synchronized many to one traffic pattern but also yields poor reconstruction performance to eliminate the transmission bottleneck of replacement node. This paper incorporates PUSH type transmission to node reconstruction, where the reconstruction procedure enables surviving nodes to accomplish reconstruction task in a pipeline manner. Each surviving node combines its local block with intermediate block with another surviving node to partially generate an intermediate block forwarded to a subsequent node. Thus PUSH can speed up reconstruction process by maximizing utilization of both network and input/output bandwidth of all surviving nodes. The PUSH based reconstruction scheme is extended for heterogeneous erasure coded storage clusters by taking into account both load and heterogeneity of surviving nodes.**

***Index Terms*— Heterogeneous erasure coded storage clusters, PULL type transmission, PUSH type transmission**

I. INTRODUCTION

A key design goal of erasure-coded storage clusters is to minimize reconstruction time, which in turn leads to high reliability by reducing vulnerability window size. PULL-Rep and PULL-Sur are two existing reconstruction schemes based on PULL-type transmission, where a rebuilding node initiates reconstruction by sending a set of read requests to surviving nodes to retrieve surviving blocks. To eliminate the transmission bottleneck of replacement nodes in PULL-Rep and mitigate the extra overhead caused by noncontiguous disk access in PULL-Sur, we incorporate PUSH-type transmissions to node reconstruction

The passive pull model inevitably encounters a transmission bottleneck problem that lies in rebuilding nodes. This project represents two PUSH-based reconstruction schemes PUSH-Rep and PUSH-Sur to improve reconstruction performance in a distributed storage cluster.

II. MOTIVATIONAL FACTORS FOR PUSH TECHNIQUES

The following three factors motivated to propose the PUSH-based reconstruction technique for erasure-coded clustered storage.

- The high cost-effectiveness of erasure-coded storage, i.e erasure-coded storage clusters have increasingly become a cost-effective and fault-tolerant solution for archive storage data centers cloud storage and the like. Especially, Reed-Solomon (RS) codes are widely used in storage clusters to provide high data reliability. For example, Windows Azure Storage (WAS) adopts a variant of RS codes to implement a four-fault-tolerant cluster system.
- The severe impact of recovery time on reliability, erasure-coded storage clusters should protect against data loss caused by node failures, because high reliability is an indispensable requirement for building large-scale storage systems. The mean-time-to data- loss or MTTDL of a r-fault-tolerant storage system is inversely proportional to the r^{th} power of recovery time of a storage node.
- The deficiency of PULL-based reconstruction I/Os, Traditional reconstruction techniques in storage clusters advocate the pull model, where a master node initiates reconstruction by sending requests to worker nodes dedicated to the reconstruction process. The passive pull model inevitably encounters a transmission bottleneck problem that lies in rebuilding nodes. This incurs TCP in cast problem.
-

Erasure Code

Erasure code is a method of data protection in which data is broken into chunks; the broken data are expanded and

).

encoded with redundant data price and store across set of different location. The broken data chunks are reassembled to re build file in case of data loss. Erasure coded heterogeneous cloud integrates components from multiple vendors [5]. Many organization uses heterogeneous storage cluster to support its business process retaining integrity of data. Organization move to heterogeneous cluster storage as it varies from one vendor to another.

III. RELATED WORK

RAID concept is used to avoid data loss from device failure. Mirroring, a popular solution, is too expensive over time. This paper presents a compromise solution that uses multi level redundancy coding to reduce the probability of data loss from multiple simultaneous device failures. This approach handles small-scale failures of one or two devices efficiently while still allowing the system to survive rare-event, larger scale failures of four or more devices and protect against rare event failures [1]

A vast majority of existing reconstruction techniques are optimized for disk arrays or Redundant Array of Inexpensive Disks (RAID). These reconstruction approaches can be classified into four categories:

- Maximizing recovery parallelism. SOR creates a number of reconstruction processes associated with strips; DOR makes every surviving disks busy with reconstruction reads at all time;
- Reducing interference between reconstruction and user I/Os. Work Out speeds up the recovery process by outsourcing all write requests and popular read requests to a surrogate RAID set
- Optimizing decoding operations. A code-specific hybrid reconstruction algorithm to reduce XOR operations and improve decoding performance during recovery
- Minimizing the number of reads on surviving disks. RDOR recovers a failed disk in a RDP-coded RAID with the decreased number of disk reads MICRO utilizes storage cache and RAID controller cache to diminish the number of reconstruction I/Os.

PUSH type transmission to node reconstruction, where the reconstruction procedure enables surviving nodes to accomplish reconstruction task in a pipeline manner. Each surviving node combines its local block with intermediate block with another surviving node to partially generate an intermediate block forwarded to a subsequent node. Thus PUSH can speed up reconstruction process by maximizing utilization of both network and input/output bandwidth of all surviving nodes. The PUSH based reconstruction scheme is extended for heterogeneous erasure coded storage clusters by taking into account both load and heterogeneity of surviving nodes..

IV. PULL-BASED RECONSTRUCTION APPROACHES

The PULL - based reconstruc -tion can be envisioned as a master-worker computing model, in which a master triggers a reconstruction procedure by sending a set of read requests, and then waits for the requests to be completed by workers. There are two classical reconstruction approaches in real-world erasure coded storage clusters:

i) a designated master (e.g., a replacement node) fetches k surviving blocks and reconstructs a failed block and ii) each surviving node plays the role of a master (i. e., acting as a rebuilding node) and all surviving nodes perform as workers, where write I/Os of rebuilt blocks are spread out over all the surviving nodes. From the angle of message communication, this 'Master- Worker' pattern belongs to the category of PULL-type transmission. Throughout this paper, we refer to the reconstruction scheme using replacement nodes as PULL-Rep; we term the solution of distributing reconstruction load among surviving nodes as PULL-Sur

PULL-Rep: PULL-Rep A replacement node fetches k Surviving blocks and rebuilds the corresponding failed block.

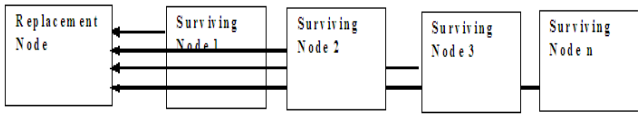

Fig 1 PULL-rep reconstruction

PULL-Sur: each surviving node fetches k - 1 surviving blocks and rebuilds the corresponding failed blocks. After the entire reconstruction process is completed, all rebuilt blocks are migrated to a new node.

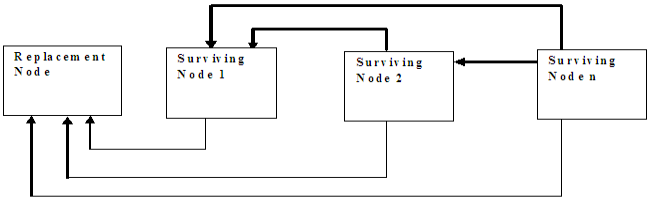

Fig 2. PULL-sur reconstruction

V. PUSH-BASED RECONSTRUCTION APPROACHES

PUSH aims to alleviate the reconstruction performance bottleneck caused by a replacement node's network bandwidth in PULL-Rep. Second, PUSH also aims to mitigate extra seeking times induced by the non-sequential disk accesses in PULL-Sur. In comparison to surviving nodes that passively respond to reconstruction reads in PULL, the surviving nodes in PUSH proactively participate in the entire reconstruction process In PUSH-Rep and PUSH-Sur, each surviving node first combines a locally stored block with a block received from another node to produce part of a final block, and then delivers the resulting intermediate block to a subsequent node. In doing so, the entire surviving node can devote all their resources, including CPU time, I/O capacity and network bandwidth, to the reconstruction process. Conceptually, a surviving node is an object-based storage device that can semi-independently manipulate its stored data.

PUSH-Rep, A surviving node creates an intermediate block using linear combination based on what it owns and receives, sends the resulting intermediate block to a subsequent node;

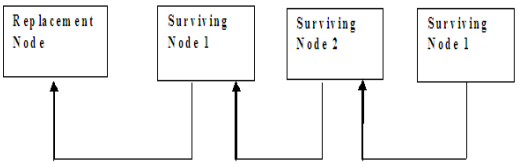

Fig 3. PUSH –rep reconstruction

PUSH-sur, PUSH-Sur: This scheme is similar to PUSH-Rep except that allthe surviving nodes are concurrently reconstructing block**s**

Design Model

Data owner uploads data into Main Cloud Server(MCS) and user downloads data from MCS .Once the data is uploaded to MCS , data is encrypted ,converted to binary format and split into four parts[2].MCS stores first and second part in google drive and third and fourth part in drop box.User sents request to main cloud server to download data and it forwards request to third party auditor to check for data integrity. [Fig 4]

XOR retrives data partition from heteroginious cloud(google drive & drop box)and performs XOR-operation . The result of XOR-operation is stored in separate cloud. Once the data are stored in the corresponding data servers the parity bits are added to the data, so that the data will be changed. Erasure Code by using the XOR operation are applied on changed data, the data will be converted in binary data while performing XOR operation on the block data.

Main cloud sever will retrieve lost data in case of cloud server and replica server failure (node failure). Main cloud server fetches the XORed data from XOR-server and reconstructs the lost data using push –rep technique.

Third party auditor intimates main cloud server in case of data loss in replica server as well as in clouds server, thereby main cloud server uses erasure code reconstructs lost data from XOR–server [4].Once added the parity added bits, then the data will be given to the Trusted Parity auditor. The Trusted Parity Auditor will generate the signature using change and response method. The data will be audited in this module, if any changes occur it will provide the intimation regarding the changes. XOR-server aggregates the data stored in different storage location to perform XOR operation

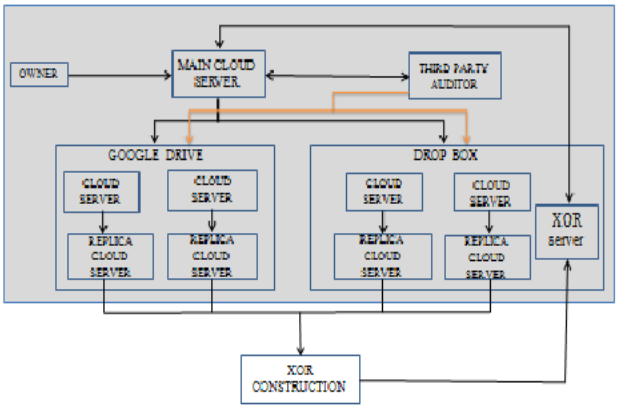

Fig 5: Architecture Diagram

VI. MODULE DESCRIPTION

Cloud Network Construction

Main Cloud Server is the area where the user requests the data and the data owner will upload their data. Once the user send the request regarding the data they want, the request will first send to the Cloud Server and the Cloud Server will forward user request to the data owner. The data Owner will send the data to user via Cloud Server. The Cloud Server will also maintain the Data owner and Users information in their Database for future purpose. Data base contains details like username, password and a set of random numbers.

Four cloud server and Replica Cloud server is maintained in heterogeneous cloud. If suppose the data in the cloud server is lost, then the Main Cloud server will contact the Replica Cloud server to retrieve lost data from the Replica Cloud Server.

Data owner registration

Data Owner uploads the data in Cloud Server. To upload the data into the Cloud server, the Data Owner have be registered in the Cloud Server. Once the Data Owner registered in cloud server, the space will be allotted to the Data Owner.

Fig 4:Data owner registration

Data upload

Once the data is uploaded into the main cloud server, the main cloud server will split the data into many parts and store all the data in the separate data servers located in heterogeneous cloud. To avoid from hacking process the data is split, encrypted and stored [2] in corresponding data server of heterogeneous cloud

Encryption [3] is done based on encryption keys that are stored in appropriate key servers, So that the security of the cloud network can be increased. Users have to provide the entire keys that are stored in the appropriate key servers to retrieve data. Once the data are stored in the corresponding data servers the parity bits are added to the data, so that the data will be changed. Erasure Code by using the XOR operation are applied on changed data, the data will be converted in binary data while performing XOR operation on the block data.

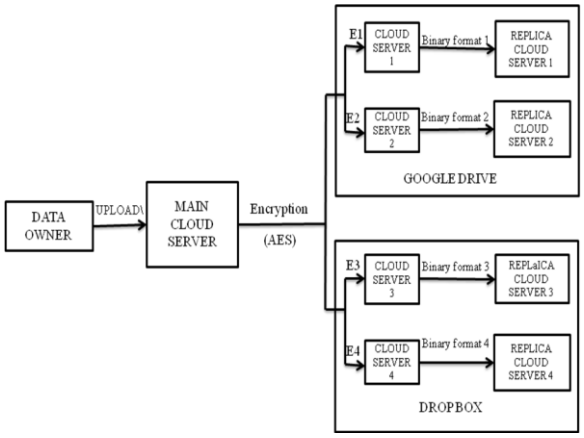

Fig 6: Data upload

XOR- construction

Once the data are stored in the corresponding data servers the parity bits are added to the data, so that the data will be changed. Erasure Code by using the XOR operation are applied on changed data, the data will be converted in binary data while performing XOR operation on the block data.

Main cloud sever will retrieve lost data in case of cloud server and replica server failure(node failure). Main cloud server fetches the XORed data from XOR-server and reconstructs the lost data using push –rep technique.

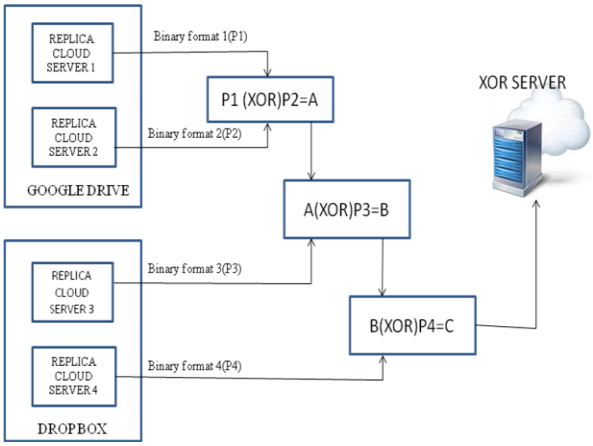

Fig 7:XOR-construction

Data Integrity Verification

User requests the main cloud server to download data uploaded by the data owner. Main cloud server requests third party auditor to verify whether data requested by users is contained in cloud server or replica server.

Third pary auditor responses to main cloud server that there is no data loss in storage cluster due to node failure ,then MCS retrieves data from storage cluster .Main cloud server retrieves data from replica server in case of data loss in cloud server.

Third party auditor intimates main cloud server in case of data loss in replica server as well as in clouds server, thereby main cloud server uses erasure code reconstructs lost data from XOR–server.

Once added the parity added bits, then the data will be given to the Trusted Parity auditor. The Trusted Parity Auditor will generate the signature using change and response method. The data will be audited in this module, if any changes occur it will provide the intimation regarding the changes. XOR-server aggregates the data stored in different storage location [5] to perform XOR operation

Reconstruction

PUSH based reconstruction scheme is used to reconstruct lost data. PUSH–Rep Reconstruction occurs using Replacement Nodes where rebuilt blocks are sequentially written to the disks of replacement nodes. PUSH–Sur allows each surviving node to rebuild a subset of failed data, so all the surviving nodes accomplish the reconstruction in parallel. After reconstruction of lost data the TPA (Trusted Party Auditor) checks the reconstructed data to maintain data integrity.

PUSH- based reconstruction techniques speeds up the reconstruction process by maximizing the utilization of input/output bandwidth of surviving nodes in heterogeneous erasure coded storage cluster. Reconstruction is performed in case of node failure i.e. a portion of partition is inaccessible by main cloud server [4].

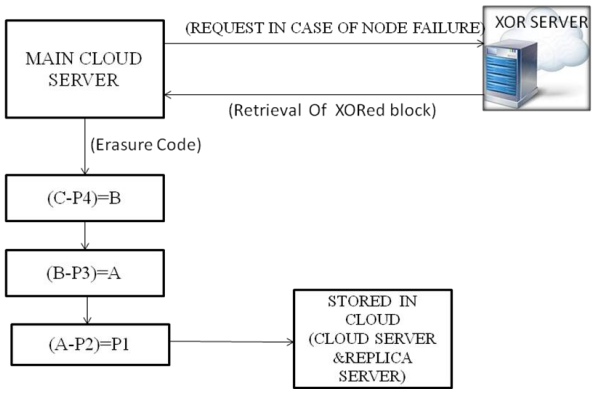

Fig 8: Reconstruction of lost data

VII. CONCLUSION AND FUTURE WORK

PUSH based reconstruction scheme is extended for heterogeneous erasure coded storage clusters. The PUSH technique reduces the reconstruction time to reconstruct lost data due to node failure. Compared to the PULL-based reconstruction technique, in which surviving blocks are transferred in a synchronized ‘M:1’ traffic pattern where as in PUSH-based reconstruction solutions support the ‘1:1’ pattern, which naturally solves the Incast problem.

Nowadays a grand challenge for storage clusters is efficiently migrating data replicas to create an erasure-coded archive. To take this challenge, the PUSH - type transmission is integrated into the archival migration in erasure-coded storage clusters. Moreover, the PUSH-based reconstruction schemes are extended for heterogeneous erasure-coded storage clusters by taking into account both load and heterogeneity of surviving nodes. PUSH-based

reconstruction technique can be extended to reconstruct lost data due to failure of more than one node in erasure storage cluster.

.

REFERENCES

[1] Avani Wildani, Thomas Schwarz and Ethan Miller proposed "Protecting against Rare Event Failures in Archival Systems" in 2009 Storage Systems Research Center Baskin School of Engineering University of California, Santa Cruz Santa Cruz, CA 95064 http://www.ssrc.ucsc.edu/.

[2] Hugo Krawczyk proposed "Cryptographic Extraction and Key Derivation: The HKDF Scheme" ' in IEEE INFOCOM' 2010.

[3] Hussam Abu-Libdeh and Lonnie Princehouse proposed "RACS: A Case for Cloud Storage Diversity" ,"IEEE Transactions on Information Forensics and Security, Vol. 7, No. 2,2010

[4] Michael Vrable and Stefan Savage proposed "Cumulus: File system Backup to the Cloud" ACM Transaction on storage volume 5, Issue 4, December 2011

[5] Sotirios Damouras ,and Phillipa Gill proposed "how to protect against latent sector errors" IEEE transaction on storage volume 6, Issue September 2010 article No.9.

D.Durai Kumar received the B.E degree in computer science and engineering from Madras University in 2004. He has received M.tech degree in computer Science and engineering from Dr.MGR university in 2009. He is currently working as associate professor and head in department of information technology at Ganadipathy tulsi's jain engineering college,Vellore. His research interests includes cloud computing, data mining, soft computing .

A.Aruna Devi received the B.E degree in Information science from visveswaraya Technological University,Bangalore,in2012. She is currently pursuing M.tech in Department of information technology from anna university , Vellore. She has published 3 technical papers in international journals and conferences Her research interests include cloud computing.