Imperial Journal of Interdisciplinary Research (IJIR)
Vol-3, Issue-3, 2017
ISSN: 2454-1362, http://www.onlinejournal.in

A Cost Effective Mechanism for Cloud Data Reliability with Minimum Replica Checking

Pallavi Diliprao Mogal
Department of Computer Engineering
Matoshri College of Engineering and Research Centre, Nashik

Abstract: *Now a day's Cloud Computing is use to store data and for resource sharing. Cloud is a new way to store large amount of data. Cloud Provides space to store data and sharing this data with multiple users. Also it is method for pay as per we use. In current cloud computing environment large scale cloud based application have put forward higher demand for storage ability of data centers. The two main concerns in current cloud storage systems are Data reliability and Storage Cost. In order to reduce the cloud storage usage while meeting the data reliability requirement, the cost-effective data reliability management generalized Proactive Replica Checking for Reliability(PRCR) mechanism is proposed. Comparing with the typically 3-replica replication strategy, PRCR can reduce the storage space consumption by one-third to two-third, reduce the storage cost. PRCR pro-actively checks the availability of replicas using DOS attack for maintaining data reliability. In PRCR data in the cloud is managed in different types according to its expected storage duration and reliability requirements. The policy to access the stored data can be maintained by using Attribute based encryption algorithm(ABE).*

Keywords: *ABE (Attribute based encryption), Cost-effective storage, Data Reliability, PRCR (Pro-actively Replica Checking for Reliability).*

1. Introduction

NOWADAYS Cloud computing has become popular among various fields like information technology and various business enterprises because in cloud user has to pay only according to their usage. Cloud makes the use of cloud service providers which provides various services to their users according to their need. Cloud computing has become one of the most popular paradigms in both academia and industry. Due to rapid progress of technology and large scale use of internet has resulted in generation of massive amount of data and cloud storage is expanding in dramatic speed. The data which are stores in cloud should first met the data reliability with high efficiency and cost-effectiveness. High reliability is critical for current cloud storage systems. However, achieving it at a reasonable cost can be a challenge, especially in large-scale systems.

In order to reduce the cloud storage usage while meeting the data reliability requirement, the generalized PRCR mechanism is proposed in which data reliability with variable disk failure rate is well investigated. To minimize replica and storage consumption in cloud, the cost-effective data reliability management mechanism named Proactive replica Checking for Reliability(PRCR) is presented. Comparing with the typically 3-replica replication strategy, PRCR can reduce the storage space consumption by one-third to two-third, hence reduce the storage cost. PRCR pro-actively checks the availability of replicas using DOS attack for maintaining data reliability. In PRCR data in the cloud is managed in different types according to its expected storage duration and reliability requirements. The policy to access the stored data can be maintained by using ABE algorithm.

1.1 Proactive replica checking

The main idea to use proactive replica checking is to propose a cost effective data reliability management mechanism for cloud data storage. The PRCR use well-known property of exponential distribution called the memory less property. In PRCR the data in the cloud is managed in different types according to its expected storage duration and reliability requirement. The data which is only for short term storage for this single replica is enough to supply the data reliability. The data which is for long term use can have higher data reliability then reliability assurances of a single replica, so two replicas are stored which are periodically and proactively checked. The proactive replica checking is done periodically to check the existences of data replicas. This task must always be done before the reliability assurances drops below the reliability requirement. If any single replica loss is found in cloud then it is quickly recover according to certain strategy like efficient replica maintenance for distributed storage system and energy-efficient data transfer strategy with link rate control for cloud. If in some extreme case both the replica may be lost, then the probability of such a situation is already incorporated in data reliability model. The PRCR ensures that the data loss rate is no more than 0.01 per cent of all the data per year.

2. Literature survey

S. Ramabhadran, C.Huang, Y. Xu[6] have studied and assumed an exponential data reliability model, in which the failure rate of each disk is a constant. For example, recent studies that analyze data reliability with Markov chain models assume that the failure rates of all disks in the storage system are the same. However, a constant disk failure rate cannot explain all of the phenomena happening in reality. It has been very well known that the failure rate of disk drives follows what is often called a "bathtub" curve, where disk failure rate is higher in the disk's early life, drops during the first year, remains relatively constant for the remainder of the disk's useful lifespan and rises again at the end of the disk's lifetime.

IDEMA, J. G. Elerath, Q.Xin, E. Pinheiro [10],[13]shows that the disk failure probability of populations of disks generally do not follow an exponential distribution. For fixing this inconsistency, the International Disk Drive Equipment and Materials Association (IDEMA) proposed a compromised presentation for disk failure rates that uses discrete disk failure rates where the lifespan of each disk are divided into different life stages with different failure rates. Such model has been demonstrated to be feasible in, and a nine-month investigation conducted by Google also obtained results very consistent to this model. In this paper, we describe the disk failure rate pattern in the IDEMA style, which divides the lifespan of disks into discrete life stages with discrete disk failure rates, and we also conduct our research based on the disk failure rates provided by IDEMA standards and Google's ninemonth disk failure trend study. Apart from research on disk reliability, many efforts for ensuring data reliability have also been made in the software aspect.

A.Gharaibeh, B.Balasubramanian[4] Among all the existing approaches for supporting data reliability, data replication has been considered as a dominant approach in current distributed storage systems. Some existing works on large-scale distributed storage systems have been proposed. Specifically, in Cloud computing, data replication technologies have been widely adopted in current commercial Cloud systems. Some typical examples include Amazon Simple Storage Service (Amazon S3), GFS, Hadoop distributed file system.

Amazon[8], Although data replication has been widely used, there is a side effect that it would consume considerable extra storage resources and incur significant additional cost. To address this issue, Amazon S3 published its reduced redundancy storage (RRS) solution to reduce the storage consumption. However, such cost reduction is realized by sacrificing data reliability. By using RRS, only a lower level of data reliability can be offered. In addition to RRS, some of our previous works made contributions in reducing storage consumption for replication-based Cloud storage.

W. Li, Y. Yang[7] proposed a cost-effective dynamic data replication strategy for data reliability in Cloud data centers, in which an incremental replication method is applied to reduce the average replica number while meeting the data reliability requirement. However, for long-term storage or storage with a very high reliability requirement, this strategy could generate even more than three replicas for the data, so its ability to reduce storage consumption is limited.

K. v. Rashmi[1], D.Gu an erasure coded storage system named Hitchhiker was proposed and implemented in HDFS. Unlike direct data replication approaches for storage, erasure coding approaches divide data into several data blocks and store them with additional erasure coding blocks. By using erasure coding approaches, data reliability can be ensured at a quite high level with very low data redundancy. However, the major disadvantage of erasure coding approaches is apparent, i.e., the computation overhead for coding and decoding data can be significant, erasure coding is not the best solution for the data storage of certain data intensive Cloud applications.

3. System Overview

The PRCR is running on virtual machines in the cloud. This virtual machine is responsible for running user interface, PRCR node and conducting proactive replica checking respectively. The user interface is responsible for determining the minimum replica number. It also creates the replica if necessary and distributes metadata of files. Whenever the original replica of file is created or uploaded in the cloud it determines the minimum number of replica according to storage duration that is for short term or long term(i.e. one or two replica) . If single replica is not enough for some file to store the data the user interface calls the cloud service to create second replica for that file. If second replica is created then for that file metadata is created and distributed to an appropriate PRCR node. The metadata have total six types of attributes file ID, time stamp, data reliability requirement, expected storage duration, checking interval and replica address. The file ID and replica address is automatically given when the original or second replica of the file are created.

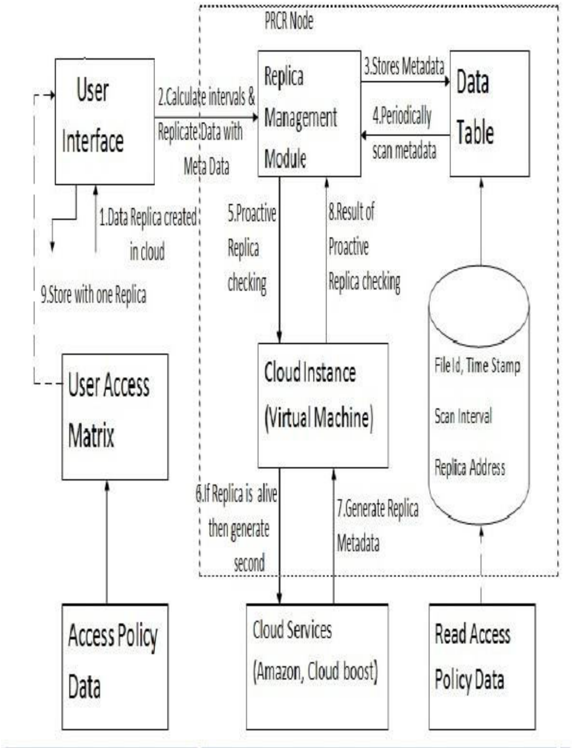

Figure 1. System Architecture

The PRCR node is responsible for management of metadata and replica. There is only one user interface and many PRCR nodes to provide the data reliability requirement with different storage duration. PRCR node can be created and destroy easily as it works independently. Each PRCR node contains tow main components: data table and replica management module. The data table maintains the PRCR node metadata attributes. The replica management module scans periodically all metadata of the file to maintain the data reliability. The data table scans the metadata to determine whether replica checking is necessary or not. In data table each round of scan is called scan cycle and this scan cycle is set to fixed value. The replica management module scans the metadata in the data table and co-operating with the cloud virtual machines to process the proactive replica checking task. The replica management module keeps on monitoring the data table to determine whether file needs to be checked or not. If any file needs to be checked then replica management obtains its metadata form data table and send it to cloud virtual machine. Cloud virtual machine does some required process and sends this data file to replica management module. After receiving file from virtual machine then replica management module conduct further actions according to the returned result. The Replica management module initializes the recovery process for creating new replica if any replica is lost.

3.1 Working Process of Proactive replica checking for reliability

1. The process starts at the time when the original replica of the file is created in the Cloud. According to the disk failure rate, the expected storage duration and data reliability requirement, the user interface determines whether to store the file with one replica or two.
2. According to the calculation in the user interface, if one replica cannot satisfy the data reliability and storage duration requirements of the file, the user interface creates a second replica by calling Cloud services, and calculates the checking interval(s) of the file. Its metadata is then distributed to the appropriate PRCR node (2). If one replica is sufficient, only the original replica is stored and the metadata of the file is not created (9).
3. Metadata attributes of the file are stored in the data table of the corresponding PRCR node.
4. Metadata is scanned periodically according to the scan cycle of the PRCR node. According to file's time stamp and the current checking interval, PRCR determines whether proactive replica checking is needed.
5. If proactive replica checking is needed, the replica management module obtains the metadata of the file from the data table.
6. The replica management module assigns the proactive replica checking task to one of the Cloud virtual machines for proactive replica checking. The Cloud virtual machine executes the task, in which both replicas of the file are checked.
7. The Cloud virtual machine conducts further action according to the result of the proactive replica checking task: if both replicas are alive or lost, go to step 8; if only one replica is lost, the virtual machine calls the Cloud services to generate a new replica based on the replica that is alive.
8. The Cloud virtual machine returns the result of the proactive replica checking task, while in the data table, the time stamp and checking interval(s) are updated. Specifically, step(1) if both replicas are not lost, the next checking interval is put forward as the current checking interval; and step(2) if a replica is lost and recovered on a new disk, the new replica address is stored and all the checking interval(s) are recalculated. Otherwise, further steps could be

conducted, for example, a data loss alert could be issued.

9. Steps 4 to 8 form a continuous loop until the expected storage duration is reached or the file is deleted. If the expected storage duration is reached, either the storage user could renew the PRCR service or PRCR could delete the metadata of the file and stop the proactive replica checking process.

4. System analysis

Proactive replica checking for reliability use two main key algorithms which are the storage duration algorithm and metadata distribution algorithm in order to determine the minimum replica number and improve the performance of PRCR. The Attribute based algorithm is also used to offers security and access control. The system also continuously keeps on monitoring the Denial of service attack in order to maintain the data reliability. The Storage duration predication algorithm has two main processes, first it determines the minimum replica number for meeting the data reliability requirement. Second give a certain data reliability requirement; it calculates the longest storage duration of the data while the data reliability requirement is met.

.

5. Conclusion

The generalized PRCR data reliability model gives the cost effective reliability management mechanism. PRCR reduces the cloud storage space by storing the minimum number of replicas of file. The stored replica are no more than two and stores according to its storage duration. Thus for short-term storage duration only one replica is enough for data reliability assurance. And for long-term storage duration two replica are created to provide the higher data reliability than short-term storage duration. PRCR periodically scans the data table to determine replicas existence and also dynamically checkes DOS Attack. If its sees that attack is going to happens then it immediately stores the data to other replica and update the changes. As user can access any data which is stored on cloud, to prevent privacy ABE is use to provide access policy to user for accessing the cloud data.

The proposed system address the server migration issue of server migration start went on attack. So the future work focuses on solving this issue.

6. Acknowledgements

I am thankful to my guide Prof.N. L. Bhale for his guidance and encouragement. Their expert suggestions and scholarly feedback had greatly enhanced the effectiveness of this work. I would like to express the deepest appreciation to authors Wenhao Li, Yun Yang, Dong Yaun, and j.Chen for their beneficial information and knowledge.

References

[1] K. V. Rashmi, N. B. Shah, D. Gu, H. Kuang, D. Borthakur, and K.Ramchandran, A hitchhikers guide to fast and efficient data reconstruction in erasure-coded data centers, in Proc. Conf.SIGCOMM, 2014, pp. 331342.

[2] W. Li, Y. Yang, and D. Yuan. 2013, Oct. 11. An energy-efficient data transfer strategy with link rate control for Cloud, Int. J. Auton. Adaptive Commun. Syst. [Online]. Available: http://www.ict.swin.edu.au/personal/yyang/papers/IJAACS-Li.pdf

[3] D. Yuan, Y. Yang, X. Liu, W. Li, L. Cui, M. Xu, and J. Chen, ”A highly practical approach towards achieving minimum datasets storage cost in the cloud,” IEEE Trans. Parallel Distrib. Syst.,vol. 24, no. 6, pp. 1234-1244, Jun. 2013.

[4] B. Balasubramanian and V. Garg, Fault tolerance in distributed systems using fused data structures, IEEE Trans. Parallel Distrib. Syst., vol. 24,no. 4, pp. 701715, Apr. 2013.

[5] W. Li, Y. Yang, J. Chen, and D. Yuan, ”A cost-effective mechanism for cloud data reliability management based on proactive replica checking,” in Proc. Int. Symp. Cluster, Cloud Grid Comput., 2012, pp. 564-571.

[6] C. Huang, H. Simitci, Y. Xu, A. Ogus, B. Calder, P. Gopalan, J. Li and S.Yekhanin, ”Erasure coding in Windows Azure storage,” in Proc. USENIX Annu. Techn. Conf., 2012, pp. 2-13.

[7] W. Li, Y. Yang, and D. Yuan, ”A novel cost-effective dynamic data replication strategy for reliability in cloud data centres,” in Proc. Int.Conf. Cloud Green Comput., 2011, pp. 496-502.

[8] Amazon. (2011). Amazon Simple Storage Service (Amazon S3). [Online]. Available: http://aws.amazon.com/s3/, 2011.

[9] R. Bachwani, L. Gryz, R. Bianchini, and C. Dubnicki,”Dynamically quantifying and improving the reliability of distributed storage systems,” in Proc. IEEE Symp. Rel. Distrib. Syst., 2008, pp. 85-94.

[10] E. Pinheiro, W. Weber, and L.A. Barroso, ” Failure trends in a large disk drive population, ” in Proc. USENIX Conf. File Storage Technol.,2007,pp.17-29.

[11] B. G. Chun, F. Dabek, A. Haeberlen, E. Sit, H. Weatherspoon, M. F. Kaashoek, J. Kubiatowicz, and

R. Morris, ”Efficient replica maintenance for distributed storage systems,” in Proc. Symp. Netw.Syst. Des. Implementation, 2006, pp. 45-58.
[12] S. Ghemawat, H. Gobioff, and S. Leung, ”The Google file system,”In Proc. ACM Symp. Oper. Syst. Principles, 2003, pp. 29-43.
[13] IDEMA, ”R2-98: Specification of hard disk drive reliability”, IDEMA Standards,1998.