Enabling Data Integrity Check and Corruption Prevention in Thin Cloud Storage

N. Kamaladheepan[#1],A. Marimuthu[#2],

[#1]*MPhilComputer ScienceResearch scholar, Government Arts College, Coimbatore, Tamilnadu, India*

[#2]*Associate Professor of Computer Science, Government Arts College, Coimbatore, Tamilnadu, India*

***Abstract*-Cloud Computing has evolved and matured, it gets growing interest in the enterprise market where economic pressures are challenging traditional IT operations. Many IT organizations face inefficiency in areas like funding projects, and resource utilization. Cloud Computing is focused on addressing these issues by cutting costs through better standardization, higher utilization, greater agility, and faster responsiveness of IT services . A main concern on Cloud journey is security of the infrastructure and the information stored in the infrastructure. To support these requirements, most organizations emphasis to move from maintaining tied infrastructure to a loosely coupled service oriented model. Data integrity became one critical factor. This paper aims to give solution to protect cloud data by dividing and storing the data in encrypted form. The data integrity is checked by the client and corrupted data is regenerated. This enables high data integrity in cloud storage.**

***Key words*- Cloud computing, Data Integrity, Multi-server, Data Corruption, Regeneration**

I. INTRODUCTION

Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable and reliable computing resources (e.g., networks, servers, and minimal consumer management effort or service provider interaction.

Cloud computing isn't much a single technology as it is a combination of many storage, applications, services) that can be rapidly provisioned and released with existing technologies. The essential characteristics of cloud computing are on-demand self-service, ubiquitous network access, resource pooling, location independence, rapid elasticity and measured service. [1]

II. CLOUD COMPUTING TYPES

There are several service and deployment models for implementing cloud technology. Each has its advantages and disadvantages with significant implications for any organization researching or actively considering a cloud deployment. The service models are

(i) Cloud Software as a Service (SaaS) - Use provider's applications over a network

(ii) Cloud Platform as a Service (PaaS) - Deploy customer-created applications to a cloud.

(iii) Cloud Infrastructure as a Service (IaaS) - Rent processing, storage, network capacity, and other fundamental computing resources.[3]

The deployment models, which can be either internally or externally implemented, are

A. *Public Cloud*: A public cloud is a cloud computing model in which services, such as applications and storage, are available for general use over the Internet. Public cloud services may be offered on a pay-per-usage mode or other purchasing models. An example of a public cloud is IBM's Blue Cloud.

B. *Private Cloud:*A private cloud is a virtualized data center that operates within a firewall. Private clouds are highly virtualized, joined together by mass quantities of IT infrastructure into resource pools, and privately owned and managed.

C. *Hybrid Cloud:*A hybrid cloud is a mix of public and private clouds.

D. *Community Cloud:*A community cloud is an infrastructure shared by several organizations which supports a specific community

III. DATA INTEGRITY IN CLOUD

Content centric or information-centric protection is the main idea of data security. Taking information and making it secure so that us and certain others can see it, is obviously not a new concept. But this is where we are struggling in the digital world. Information in a cloud environment has much more dynamism and fluidity than information that is static on a desktop or in a network folder, so we need to think of a new way protect information. Data integrity in cloud requires modifications are not made to data by unauthorized personnel or authorized personnel and processes. Also the data must be internally and externally consistent. [4]

IV. CHALLENGES IN CLOUD DATA STORAGE

Cloud storage offers an on-demand data outsourcing service model, and is gaining popularity due to its elasticity and low maintenance cost.

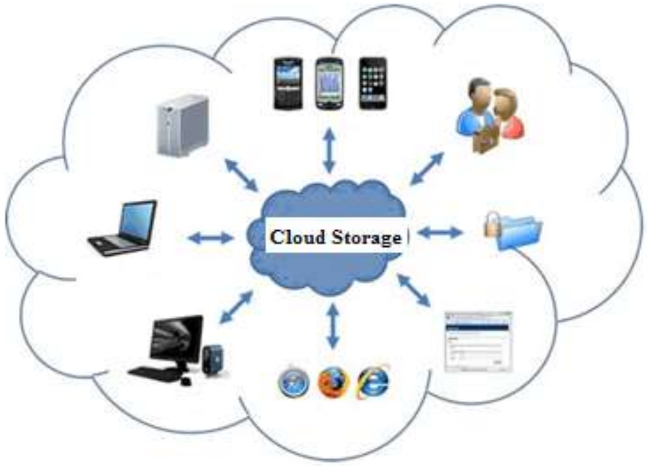

Fig.1 Cloud Storage System

However, security concerns arise when data storage is outsourced to third-party cloud storage providers. Integrity is the guarantee that the data sent is the data received and that the data is not intentionally or unintentionally altered.

It is desirable to enable cloud clients to verify the integrity of their outsourced data, in case their data have been accidentally corrupted or maliciously compromised by insider/outsider attacks. The main data attacks that will affect the cloud data are data breaches and data loss, denials of service, and malicious uses. Some of the elements used to ensure integrity are firewall services, communication security management and intrusion detection services. [1]

V. APPROACHES TO SECURE CLOUD DATA

This paper mainly deals with protecting the long term archival data in thin cloud storage. Thin cloud storage is that the client only uses the storage space of the vendor by paying to the vendor. As the data may not be accessed frequently by the client, there may be more chance of the attackers to corrupt the data in the cloud storage. A new Data Integrity Protection mechanismis proposed to protect the data in cloud.

Also the client can check the integrity of data by randomly picking up the data and checking it using Trusted Parity Auditor (TPA). TPA is an agent which is responsible for checking the originality of the client data. If the client data is corrupted, then TPA notifies the same to the client.

A regeneration erasure code mechanism is proposed to regenerate the corrupted data from the replica servers. In prior methods, solution for this problem are proposed for single server only. In single server case, if the server fails then the whole data gets lost. [2]

Suppose that we outsource storage to a server, which could be a storage site or a cloud-storage provider. If we detect corruptions in our outsourced data (e.g., when a server crashes or is compromised), then we should repair the corrupted data and restore the original data. However, putting all data in a single server is susceptible to the single point- of-failure problem and vendor lock-ins. A plausible solution is to stripe data across multiple servers. In this research work, the solution is proposed for multi-server setting in which if, data corrupted by attackers from a cloud server can be reconstructed from the secondary servers. Thus, to repair a failed server, we can

(i) Read data from the other surviving servers,

(ii) Reconstruct the corrupted data of the failed server, and

(iii) Write the reconstructed data to a new server.

In particular, erasure coding has a lower storage overhead than replication under the same fault tolerance level. In a distributed environment, an attacker chooses a specific client but the distribution of data into multiple server makes the attacker's job more difficult. Data is encrypted and divided in to chunks. If a part of data is corrupted in a server, then it is recovered from the secondary server.MR-PDP and HAIL [2] extend integrity checks to a multi-server setting using replication and erasure coding, respectively. [1]

VI. DATA INTEGRITY IN THIN CLOUD STORAGE

The basic operations to be performed for storing the file in multiple servers are

(i) Generate the secret key that are used for encrypting and decrypting the files.

(ii) Encode the file F of size |F| into n pieces of size |F|/k each, where k<n

(iii) Split the file in to chunks and apply AECC for the i^{th} chunk Pi.

(iv) Apply XOR operation for $P_1 iXORP_2 i$, $P_2 iXORP_3 i$,……$P_n i$

(v) Upload the code chunks Pi's to respective servers

To maintain the integrity of data in servers, we verify random chunks chosen from the servers. Thus k (n-k) code chunks from any k servers can be decoded to the original k (n-k) native chunks, we must have rank (A) =k (n-k). Now we can pick a chunk Pi from one of the remaining n-k servers and its rank is also checked. If it is not consistent, then we have to reconstruct the data. [1]

Download the corrupted file, its AECC parities from the server and apply error correction. Verify the correct chunk with SHA 256 algorithm. Repeat the same for (n-k) code chunks.

If the server fails or having a large number of corrupted data, then download the chunk from all remaining n-1 servers. Again encode and upload the data to the new server.

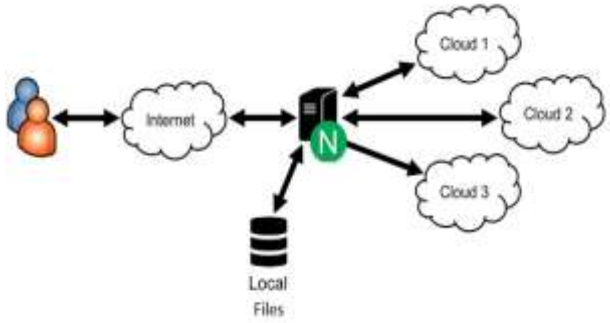

Fig.2Multi-server Environment

Based on the algorithm, the below steps are proposed to check the data integrity in cloud storage.

(i) The client who wants to store data in the cloud should register the details such as user name, password, email id, phone number.

(ii) Once the registration is completed, the client is allowed to upload file to the cloud server.

(iii) The NCC cloud connector is used to connect to the Dropbox public cloud and its space is utilized.

(iv) The file being uploaded is encrypted using Advanced Encryption Standard (AES-128 bit algorithm) and divided in to chunks, parity bit is added and stored in multiple cloud servers.

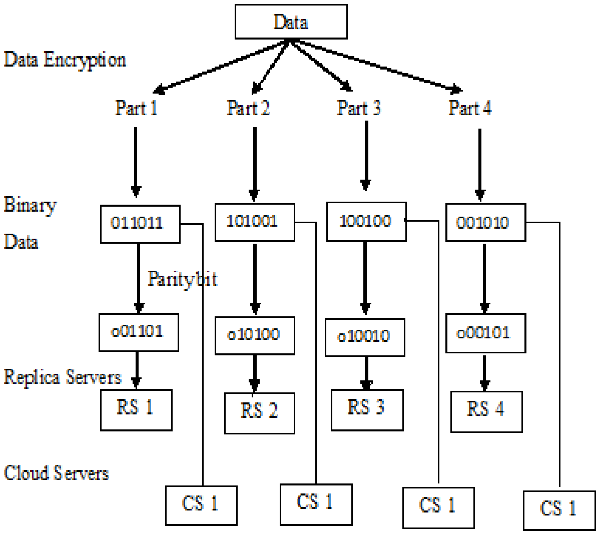

Fig.3 Parity Bit Addition and Erasure Code

(v) Trusted Parity Auditor (TPA) uses the parity bits in the primary cloud servers for integrity check.

(vi) Encrypted data without parity addition is stored in the replica servers.

(vii) Client performs the data integrity check on randomly chosen parts of data in the cloud server using TPA.

(viii)The TPA uses SHA 256 algorithm and compares the hash value of the corrupted file and the original file in the replica server. If it is not equal, then an intimation is sent to the client mail about the data corruption.

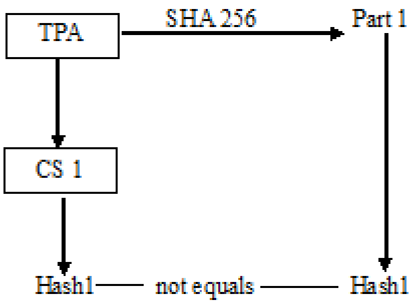

Fig.4Data Integrity Check using SHA Algorithm

(ix) While downloading the data, all the data are retrieved from the replica server without data loss.

(x) As a protective mechanism, all the parts of data are combined with XOR operation and stored in backup server in a different location. This enables high data integrity in cloud.

VII. ANALYSIS AND EVALUATION

The running time of Data Integrity codes were evaluated in the local cloud platform. We measure the running time or each operation. Our results are averaged over 10 runs. We used files of size 100MB for evaluation purposes.

Using Secure Hash Algorithm, the integrity of the data are verified. Time is consumed according to the size of the data encoded. Time taken for all data are displayed.

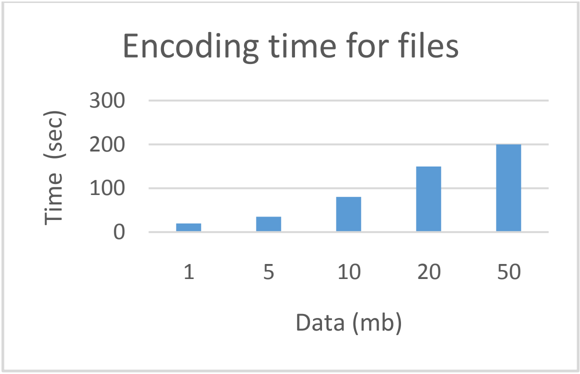

Fig.5Time taken by files for encoding

Also the ranks of different chunks are checked and evaluated. The below graph shows the time taken to check different percentage size of files.

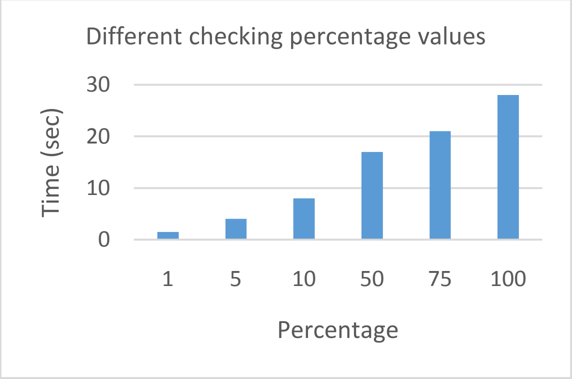

Fig.6 Different checking percentage values

VIII. CONCLUSION

In this paper, we have outlined the new approach to perform secure replication of stored information. This approach enables the client to verify the integrity of their data in cloud. This is a dominant technique which will provide better results for security and availability of data. We can use this secure replication technique in order to build a secure and reliable distributed storage. We expect the enhancement done in this technique will increase the quality by different data mart host with cloud provider and store information accordingly based on sensitivity. This new approach can be used by different cloud providers and other organizations for enabling data integrity.

REFERENCES

[1] Henry C. H. and Patrick P. C. Lee, "Enabling Data Integrity Protection in Regenerating-Coding-Based Cloud Storage", 31st International Symposium on Reliable Distributed Systems, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 25, NO. 2, FEBRUARY 2014

[2] K. Bowers, A. Juels, and A. Oprea, "HAIL: A High-Availability and Integrity Layer for Cloud Storage," Proc. 16th ACM Conf.Computer and Comm. Security (CCS '09), 2009.H. Abu-Libdeh, L. Princehouse, and H. Weatherspoon, "RACS: A Case for Cloud Storage Diversity," Proc. First ACM Symp. Cloud Computing (SoCC '10), 2010.

[3] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A View of Cloud Computing," Comm. ACM, vol. 53, no. 4, pp 50-58, 2010.

[4] Cloud Security- A comprehensive guide to secure cloud computing by Ronald L. Krutz, Russell Dean Vines, Wiley India Pvt. Ltd. Edition 2010, ISBN: 978-81-265-2809-7

[5] Cloud Computing by David cookers, Tata McGraw Hill 2012 edition, ISBN-13:978-1-25-906104-2

[6] Gupta Sarika, Sangita Rani Satapathy, Mehta Piyush and TripathyAnupam, "A Secure and Searchable Data Storage in Cloud Computing", 3rd IEEE International Advance Computing Conference (IACC), 2013, page 106-109.

[7] Taeho Jung, Xiang-Yang, Zhiguo Wan, Meng Wan, "Privacy Preserving Cloud Data Access*With Multi-* Authorities", Proceedings IEEE INFOCOM, 2013, page 2625-2633.

[8] Amazon.com. Amazon simple storage service (Amazon S3), 2008. Referenced 2008 at aws.amazon.com/s3.

[9] https://www.dropbox.com/

[10] Cloud Computing Implementation, Management, and Security by John W. Rittinghouse, James F. Ransome, Third Indian Reprint 2014 by CRC Press. ISBN 978-1-4398-0680-7

[11] Cloud Computing Principles and Paradigms edited by RajkumarBuyya, James Broberg, Andrej Goscinski, Authorized reprint by Willey India Pvt. Ltd, ISBN 978-81-265-4125-6