International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 6 | ISSN : 2456-3307

Secure Data Duplication Process for Better Performance of Primary Databases

Mohd Nadeem[1], Md Ateeq Ur Rahman[2]
[1]M.Tech Scholar, Department of Computer Science & Engineering, Shadan College of Engineering & Technology, Hyderabad, Telangana, India
[2]Professor, Department of Computer Science & Engineering, Shadan College of Engineering & Technology, Hyderabad, Telangana, India

ABSTRACT

In Real-time scenario, the data duplication is available but not dynamically implemented. The purpose of this paper is to study the data deduplication and performance specially when dealing with remote server. Normally remote servers are not capable of detecting data deduplication as they are situated and programmed in such a way that their job is to accept the data from many users around the globe. At client side only, we can implement a technique or scheme where data deduplication can be detected and informed to the data owner to save cloud infrastructure. To Detect data being duplicated in cloud servers for more resources availability and fast performance. Storage data on remote servers requires attention on both security and consistency. The data owner can check and verify their data stored in duplicates in cloud server before uploading any new content from the client side. By introducing a new and novel technique this paper achieved the goal of detecting and instructing data duplication in cloud server before outsourcing.

Keywords: Data Duplication, Deduplication, iDedup, Select iDedup, SecureDeDup, Data Blocks, Token number, Infrastructure as a Service.

I. INTRODUCTION

To address the issue of data deduplication there has been many schemes introduced in cloud computing. The main goal was to lower the resources cost from the cloud server. Data deduplication increases the volume of resources for data storage and limits it to consumers. As cloud computing offers vast and heavy data storage service there is a possibility of duplicate data storage in a server or multiple servers. Many techniques implemented till now focuses on security issues and sided the issue of data duplication on cloud server. To utilize the

available resources in a very official and implicit manner there is a need to verify data of an individual or an organization where data is being stored in duplicates in cloud server. Information deduplication has been exhibited to be a successful method in Cloud reinforcement and documenting applications to lessen the reinforcement window, enhance the storage room efficiency and system data transfer capacity usage. For instance, the ideal opportunity for the live VM movement in the Cloud can be significantly diminished by embracing the information deduplication innovation. The current information deduplication plans for essential storage, for example, iDedup and Offline-Dedupe, are limit arranged in that they concentrate on capacity limit reserve funds and just select the vast solicitations to deduplicate and sidestep all the little demands (e.g., 4KB, 8KB or less). The method of reasoning is that the little I/O asks for represent a small portion of the capacity limit necessity, making deduplication on them unprofitable and possibly counterproductive considering the significant deduplication overhead included. Not with standing, past workload considers have uncovered that little files command in essential stockpiling frameworks (over half) and are at the base of the framework execution bottleneck.

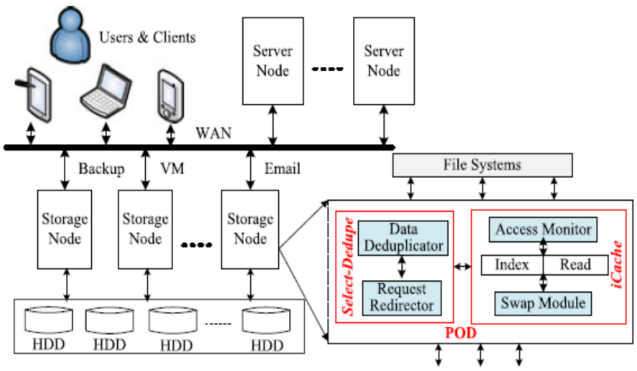


Figure 1. Proposed system framework

II. SYSTEM ANALYSIS

2.1 Introduction

We audit some related works including data based cryptosystems and access control with security gadget in this area.

Existing System:

Many schemes have been implemented to control and monitor and data deduplication process from all different perspective. But they fail to provide deduplication, security and performance simultaneously. The current information deduplication plans for essential storage, for example, iDedup and Offline-Dedupe, are limit situated in that they concentrate on capacity limit reserve funds and just select the huge solicitations to deduplicate and sidestep all the little demands (e.g., 4 KB, 8 KB or less). The method of reasoning is that the little I/O asks for represent a small division of the capacity limit prerequisite, making deduplication on them unrewarding and possibly counterproductive considering the significant deduplication overhead included. Nonetheless, past workload examines have uncovered that little documents command in essential stockpiling frameworks (more than 50 percent) and are at the foundation of the framework execution bottleneck. Moreover, because of the cradle impact, essential storage workloads show evident I/O burstiness.

2.1.1 Disadvantages of Existing System:

- Firstly, Cloud infrastructure is not computed with which cloud service providers have to maintain huge amount of resources to maintain and monitor client's data.
- As Cloud Resources are doubled cloud service providers has to implicitly charge heavy from the client.
- No fast data retrieval by any device
- No data security for multi cloud environment

Performance issues are noted in both public and private clouds.

Consumers are hesitating to store their personal data where no security and more cost has to tolerate.

The existing system works perfectly without any issues but when integrated with icache or idedupe the performance issues are concerns.

2.2 Proposed System:

To Address the issue of performance over the cloud with low cost and good security this paper proposes a novel technique SecureDedup (SD) where data deduplication is detected at client side.

SecureDedup allows the client to outsource only unique and effective data from the machine to remote server. It doesn't outsource duplicate data from same client. The probability of duplication is restricted to same client. However other clients can have and can outsource same data with different attributes. This technique detects majorly different attributes from same client like type of data being uploaded, size of the data being uploaded, content of the data being uploaded etc,

To Check and verify data deduplication every time client needs to interact with server in existing system. But in this paper, we are proposing a technique where data deduplication concept is detected at client side. Means there is no need to visit to server for a small sub sequent request all the time. As server will be limited or restricted in processing clients request simultaneously.

Security point of view client's data in stored in different chunks or blocks in multiple cloud servers. These blocks are encrypted while outsourcing to server. Here also the performance of the server is not compromised. Every block will have a token implicitly generated by the cloud server and encrypts it when needed.

The Authentication and authorization process is handled in a very timely and securely fashion. Every

time data owner login a secret key will be sent to his/her registered mail address as a second step or two factor authentications.

2.2.1 Advantages of Proposed System:

- Better Utilization of Cloud infrastructure with which cloud service providers have to maintain huge amount of resources to maintain and monitor client's data.
- Reduction in storage cost from the CSP to attract more number of clients.
- fast data retrieval by any device and data security for multi cloud environment
- Performance is very good in both public and private clouds.
- Consumers can rely on cloud servers to store their personal data where high security with low cost is available.

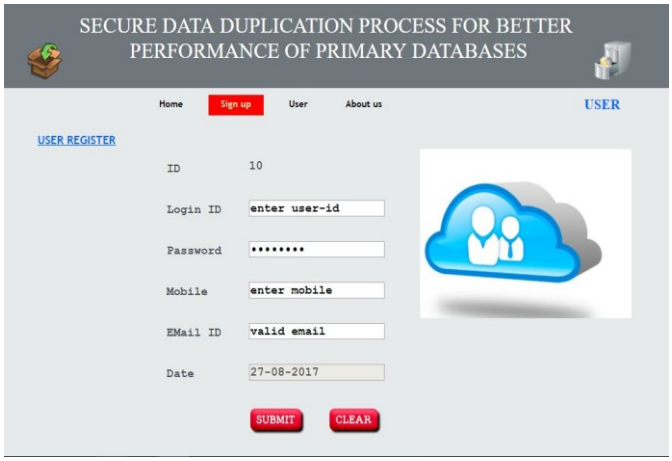
III. METHODOLOGY

To address the imperative execution issue of essential storage in the Cloud, and the above deduplication-actuated issues, we propose a Performance-Oriented Data Deduplication conspire, called POD, instead of a limit situated one (e.g., iDedup), to enhance the I/O execution of essential stockpiling frameworks in the Cloud by considering the workload attributes. Unit adopts a two-dimensional strategy to enhancing the execution of essential stockpiling frameworks and limiting execution overhead of deduplication, to be specific, a demand based particular deduplication method, called SelectDedupe, to ease the information discontinuity and a versatile memory administration conspire, called iCache, to facilitate the memory conflict between the bursty read traffic and the bursty compose traffic. All the more specifically, Select-Dedupe takes the workload qualities of little I/O-ask for mastery into the outline contemplations. It deduplicates all the compose demands if them compose information is as of now put away consecutively on plates, including the little compose demands that would somehow or another be circumvent from by the limit arranged deduplication plans. For other compose demands, Select-Dedupe does not deduplicate their excess compose information to keep up the execution of the resulting read solicitations to this information. iCache powerfully changes the reserve space parcel between the record store and the read store as indicated by the workload attributes, and swaps these information amongst memory and back-end stockpiling gadgets as needs be. Amid the compose serious bursty periods, iCache expands the file reserve size and psychologists the read store size to identify considerably more excess compose demands, in this way enhancing the compose execution. Amid the read-concentrated bursty periods, then again, the read store estimate is augmented to reserve more sweltering read information to enhance the read execution. In this way, the memory efficiency is augmented.

The model of the POD conspire is executed as an implanted module at the square gadget level and a sub file deduplication approach is utilized. To analyze the net impact of the POD plot, in our follow driven assessment we utilize the piece level follows that were gathered underneath the memory support reserve with the goal that the storing/buffering impact of the capacity stack is as of now completely caught by the follows. At the end of the day, all the little I/O asks for in our assessment are issued from the support store to the square gadgets after the previous has handled the filesystem-issued demands. The broad follow driven tests led on our lightweight model execution of POD demonstrate that POD significantly outflanks iDedup in the I/O execution measure of essential stockpiling frameworks without sacrificing the space reserve funds of the last mentioned.

IV. IMPLEMENTATION & RESULTS

Below are some of the results of the project which demonstrates step by step process of entire application.



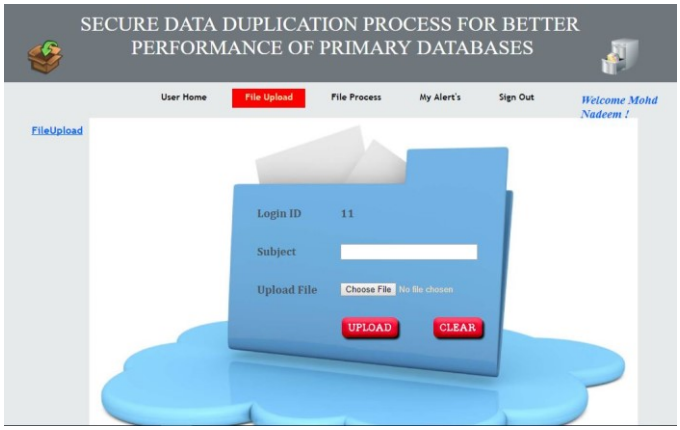
Output Screenshot 1: Registration page

In this page user can register to create an account by providing general information like username, password, email address and phone number to register. Once

registration is completed, user can sign in the application.

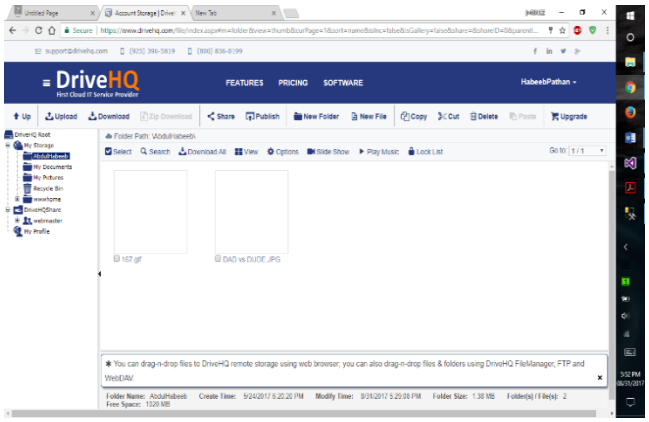
Output Screenshot 2: User Login page

Data Owner login page provides users to login in to the application and access it. User login page requires basic information to login like username, password. If the user credentials are correct, then users are authenticated.



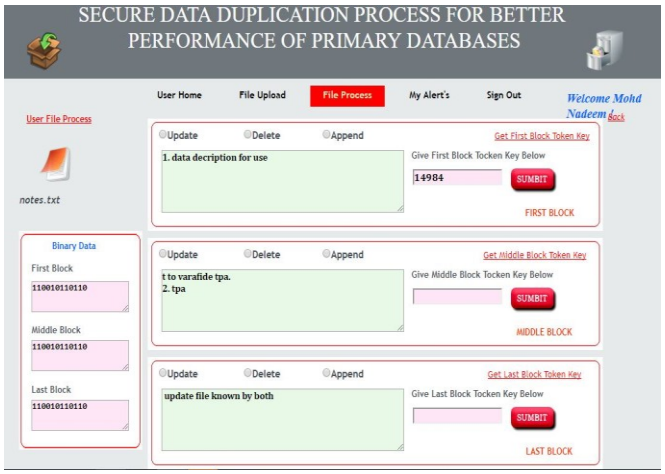
Output Screenshot 3: File upload page

After successful login in to the cloud server, file upload page is displayed to upload the file data in to the cloud server.



Output Screenshot 4: File upload in DriveHQ

The user will receive the uploaded content in cloud server.



Output Screenshot 5: file download

Here user can download their files by merging it. The download link will be provided after secure authentication in this page.

V. CONCLUSION

SecureDedup allows the client to outsource only unique and effective data from the machine to remote server. It doesn't outsource duplicate data from same client. The probability of duplication is restricted to same client. However other clients can have and can outsource same data with different attributes. This technique detects majorly different attributes from same client like type of data being uploaded, size of the data being uploaded, content of the data being uploaded etc, To Check and verify data deduplication every time client needs to interact with server in existing system. But in this paper, we are proposing a technique where data deduplication concept is detected at client side. Means there is no need to visit to server for a small sub sequent request all the time. As server will be limited or restricted in processing clients request simultaneously.

VI. REFERENCES

[1]. M. Fu, D. Feng, Y. Hua, X. He, Z. Chen, W. Xia, F. Huang, and Q. Liu. Proposed a paper Accelerating Restore and Garbage Collection in Deduplication-based Backup Systems via Exploiting Historical Information. In USENIX'14, Jun. 2014.

[2]. J. Lofstead, M. Polte, G. Gibson, S. Klasky, K. Schwan, R. Oldfield, M. Wolf, and Q. Liu. Proposed a paper Six Degrees of Scientific Data: Reading Patterns for Extreme Scale Science IO. In HPDC'11, Jun. 2011.

[3]. C. Zhang, X. Yu, A. Krishnamurthy, and Randolph Y. Wang. Proposed a paper Configuring and Scheduling

an Eager-Writing Disk Array for a Transaction Processing Workload. In FAST'02, Jan. 2002.

[4]. F. Chen, T. Luo, and X. Zhang. Proposed a paper CAFTL: A Content-Aware Flash Translation Layer Enhancing the Lifespan of Flash Memory based Solid State Drives. In FAST'11, pages 77–90, Feb. 2011.

[5]. E. Rozier and W. Sanders. Proposed a paper A Framework for Efficient Evaluation of the Fault Tolerance of Deduplicated Storage Systems. In DSN'12, Jun. 2012.