A Survey on: Load Balancing and De-Duplication in Cloud Computing

Pooja Y. Bansode[1], Payal A. Lokhande[2], Siddhant Sawant[3], Prof. R. B. Nangare[4]

[1, 2, 3] *BE Student, Computer Engineering Department, SVIT, Chincholi, Nashik, India.*
[4] *Professor, Computer Engineering Department, SVIT, Chincholi, Nashik, India.*

ABSTRACT

Now days, cloud computing is very important in the Information Technology. Cloud computing enables access to a shared pool of configurable computing resources like servers, storage and applications, etc. The storage services provided to users are though internet. Load balancing is being an important task for doing operations in cloud. And so as de-duplication also. As cloud computing has been growing and many clients all over the world are demanding more services and better results, so load balancing is necessary. Load balancing assure efficient resource utilization to customers on their demand and build up the overall performance of cloud. Every increasing volume of back up data in cloud storage may be a vital challenge. De-duplication for eliminating the duplicate data. Many algorithms have been developed for allocating client's requests to available remote nodes. The key idea behind this paper is to develop a dynamic load balancing algorithm based on de-duplication to balance the load across the storage nodes during the expansion of private cloud storage.

Keyword: - *Cloud Computing, Load balancing, Secure De-duplication, etc.*

1. INTRODUCTION

Cloud computing is an emerging on demand, internet based system. It provides variety of services over internet such as storage of data, software and hardware. "Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction". Due to rising costs, IT companies have started to externalize their IT services, which are maintained by specialized companies called service providers. This led to the emergence of cloud computing. Cloud Computing is a computing environment, where resources such as computing power, storage, network and software are abstracted and provided as services in a distributed network. This is a technology where the task is executed by sharing and using existing resources and applications of a distributed network environment.

2. LITERATURE SURVEY

Because of rise in the costs, IT companies have started to externalize their IT services, which are maintained by specialized companies called service providers. This has made the cloud computing come up. Cloud Computing is a computing environment, where resources such as computing power, storage, network and software are abstracted and provided as services in a distributed network. Cloud Computing is a technology where the job is executed by sharing and using existing resources and applications of a distributed network environment. The resources can be allocated and de-allocated with ease by the service provider. A huge number of users request services to the cloud, which is run like large internet. Various companies use cloud computing due exponential growth in users and their needs. There are cloud computing data-centers all over the world to make cloud computing feasible. Different cloud services such as pay-per-use scheme which are offered at a lower price without intervention of owner and manager of these services.

Cloud computing consist of several characteristics such as:

- On-demand- Cloud services are given on-demand. Users can get there tasks done when they want.
- Extensive Network Access- In cloud computing resources are scattered over a network .These resources are accessed through various mechanisms.

- Resource Pooling- The resources are pooled accordingly. The resources are dynamically allocated and de-allocated accordingly.
- Scalability- Quantity of resources is increase at any time according to the customer's requirements.

As it has been the norm, there are some issues with cloud computing as well. These issues come with huge number of requests that these clouds serve. Load Balancing, Redundancy and Fault tolerance are such issues. This report discusses the basics of load balancing and de-duplication.

Millions of service users across the globe constantly send service requests to the cloud for their storing or computing tasks. The cloud computing needs to provide the abstraction that the user's task is being done exclusively and provide the output without fail. When there is a surge in requests, the resource that serves these requests also needs upgrading and updating. The cloud computing has to function in such a way that it balances the load that is being out on it. A technique called load balancing is employed at this point. Cloud load balancing is the method of distributing services and computing resources in a cloud computing environment. Load balancing allows organizations to manage the workload demands by allocating resources to multiple computers, networks or servers within the cloud. By sharing the workload, the task is performed concurrently. It serves the basic idea that not all the burden should be forced on one server alone. All the servers and resources work in unison and the output is then generated in the end when all the resources have finished their assignment.

As cloud technology becomes prevalent along with it the data sharing and storage also became prevalent. The increasing volume of data needs to be managed because the less you store, the less will be the need of hardware resource. Service providers have to keep this in mind because adding hardware to store more data increases cost. To the user also it should be cheaper than actually storing the data at his end. Data de-duplication is one of the most popular technologies in storage right now because it allows companies to save a lot of money on storage costs to store the data and on the bandwidth costs. This is great news for cloud providers, because if you store less, you need less hardware. If you can de-duplicate what you store, you can better utilize your existing storage space, which can save money by using what you have more efficiently. If you store less, you also back up less, which again means less hardware and backup media. If you store less, you also send less data over the network in case of a disaster, which means you save money in hardware and network costs over time. Data de-duplicating is really a game changer and cost saver. The business advantages of data de-duplication include:

- Lowered backup costs
- Lowered hardware costs
- Lowered costs for business continuity and/or disaster recovery
- Storage efficiency
- Network efficiency

In simple words, data de-duplication compares objects (usually files or blocks) and eradicates objects (copies) that already exist in the data set. The de-duplication process deletes blocks that are not unique.

3. EXISTING SYSTEM

A number of de-duplication systems have been proposed based on various de-duplication strategies, such as client-side or server-side de-duplications, file-level or block-level de-duplication.

Bellare et. Al. showed how to protect data confidentiality by transforming the predictable message into unpredictable message. The first problem is integrity auditing. And the second problem is secure de-duplication. One of technical challenges with regards to distributed data de-duplication is to achieve scalable throughput and a system-wide data reduction ratio close to that of a centralized de-duplication system.

Following load balancing techniques are currently prevalent in clouds:

VectorDot- A. Singh et al. proposed a novel load balancing algorithm called VectorDot. It handles the hierarchical complexity of the data-center and multidimensionality of resource loads across servers, network switches, and storage in an agile data center that has integrated server and storage virtualization technologies. VectorDot uses dot product to distinguish nodes based on the item requirements and helps in removing overloads on servers, switches and storage nodes.

Join-Idle-Queue- Y. Lua et al. [14] proposed a Join-Idle-Queue load balancing algorithm for dynamically scalable web services. This algorithm provides large-scale load balancing with distributed dispatchers by, first load balancing idle processors across dispatchers for the availability of idle processors at each dispatcher and then, assigning jobs to processors to reduce average queue length at each processor. By removing the load balancing work from the critical

path of request processing, it effectively reduces the system load, incurs no communication overhead at job arrivals and does not increase actual response time.

In the existing cloud server storage techniques there is a less security for the update, delete and download file. There is less load balancing techniques and no De-duplication. The current system has only provides the spaces on the server but not avoid the duplicate files. So to avoid those problems this system is proposed.

4. PROPOSED SYSTEM

In our proposed system, in order to have a secure storage of de-duplicated data over a cloud computing we use the encryption/decryption technique. And for handling de-duplication we are using secure hash algorithm. Encryption is the process of converting a plaintext into a cipher text. Decryption is the process of converting a cipher text into plain text. The process of symmetric key encryption where same key is used for both encryption and decryption .the key ought to be send firmly over a network. To stop unauthorized access, a secure proof of possession protocol is additionally required to provide the proof that the user indeed owns the similar file when a duplicate is found. Once the proof, consequent users with the similar file are going to be provided a pointer from the server while not having to transfer the similar file. Load balancing algorithm is used to distribute the load among various nodes in the distributed system to improve the resource utilization and request response time of the system. These algorithms are mainly used to overcome the situation where a node is heavily loaded and other nodes are idle and because of which the request fails.

In this system, we have composed four modules:

- Register and login module: User have to just register his login credentials details so that he will receive the authorities to use the services. After approval by admin, user will receive his username, password and private key in his email id.
- Upload module: After login the system, user can upload his documents either it is text files or jpg file. After uploading, a confirmation message will be display to user that his file is uploaded and user can view his file. If user can upload same file with same name or different name than again a confirmation message will display that file is already uploaded so that it takes less space on cloud storage to store that particular file.
- Deletion and download modules: User can delete his own file by enter the private key. If the file has any pointer than only database entry will be deleted. If it is unique file than database entry, chucks files is deleted. User can download his file by entering the private key.
- TPI Integrity module: If some changes happen in the cloud or some hacking process is created than to check this user can scan his file. If some changes happens than tpi integrity is compressed message is displayed and if not than tpi integrity is not compressed

5. SYSTEM ARCHITECTURE

When we open the system we have go through the following functionality:

1) User can apply for login credentials.
2) **Admin**:
   - Admin will approve or disapprove the User request for login credentials.
   - If admin approves the request, then following details will send to the user.
   - Username, Password and Secret Pin to download and delete file.
   - If admin disapproves the request then user record will be deleted and disapproval mail will be sent to the user.
   - User can upload the file.
3) User can delete his own file by using secret pin.
4) On **Upload**, following operation will happen to achieve De0duplication:
   - Hash Code generation on the basis of content of the file.
   - If same hash code exists in database table then pointer will be set to the existing file.
   - If hash code is unique then file will be split into three equal chunks.
   - Then every chunk will be uploaded into three different locations.

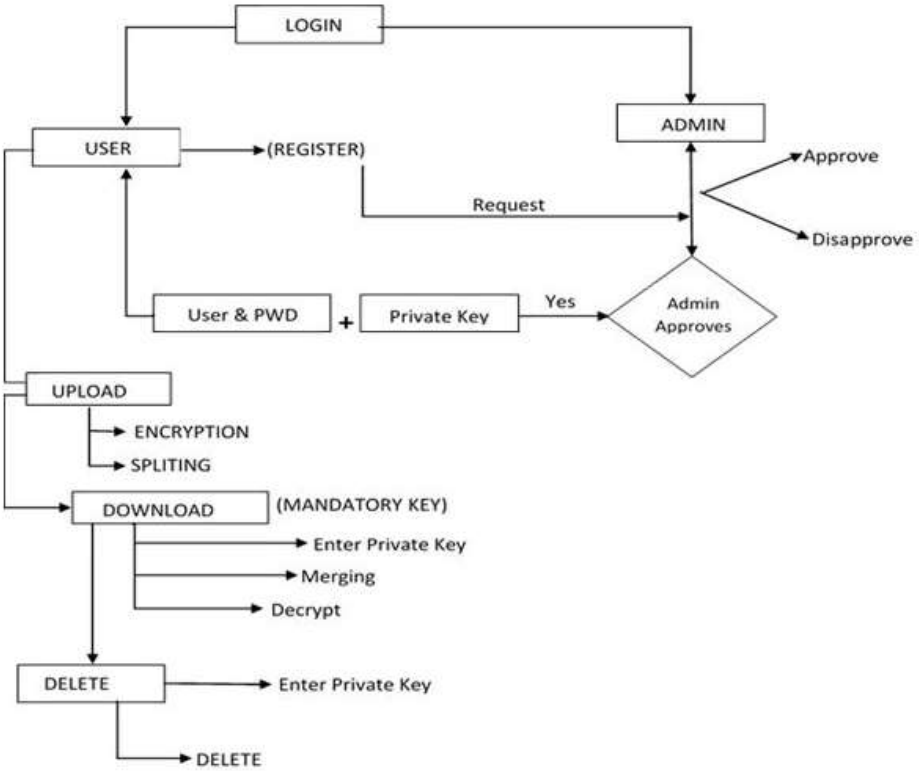

Fig -1: System Architecture Diagram.

5) On **Delete** following operation will happen:
   - User provides secret pin on rise of delete request.
   - If file has any pointer then only database entry will be deleted.
   - If there is no pointer to the file then it is a unique file and database entry and file chunks will be deleted.
6) On **Download** following operation will happen:
   - User provides secret pin on rise of download request.
   - If secret pin matched then only file chunks will be merged.
   - Then system will decrypt the file and will be downloaded to the client side.

6. CONCLUSIONS

This system proposes the architecture of de-duplication system for cloud storage environment and gives the process of avoiding de-duplication in each stage. In Client, system employ the file-level and chunk-level de-duplication to avoid duplication. The algorithm also supports mutual inclusion and exclusion. Load sharing algorithm which is having policy to partitions the system into various domains and also having concept of cache manager and information dissemination for the various cloudlets.

7. ACKNOWLEDGEMENT

We take this opportunity to express our hearty thanks to all those who helped us in the completion of the Paper. We express our deep sense of gratitude to our Project Guide Prof. M. B. Wagh, Co-Guide Prof. R. B. Nangare Computer Engineering Department, Sir Visvesvaraya Institute of Technology, Chincholi for his guidance and continuous motivation. We gratefully acknowledge the help provided by him on many occasions, for improvement of this project report with great interest. We would be failing in our duties, if we do not express our deep sense of gratitude to Prof. S. M. Rokade, Head, Computer Engineering Department for permitting us to avail

the facility and constant encouragement. Lastly we would like to thank all the staff members, colleagues, and all our friends for their help and support from time to time.

8. REFERENCES

[1] J. Wu, L. Ping, X. Ge, Y. Wang, and J. Fu, "Cloud storage as the infrastructure of cloud computing," in Proc. 2010 Int. Conf. Intell. Comput. Cognitive Inform. (ICICCI), Kuala Lumpur, 2010, pp. 380-383.

[2] J. Gantz and D. Reinsel, "The digital universe decade-Are you ready," IDC White Paper, http://www.emc.com/collateral/analyst-reports/idc-digital universe are-you-ready.pdf, 2010.

[3] P. Xie, "Survey on de-duplication techniques for storage systems," Comput. Sci., vol. 41, no. 1, pp. 22-30, Jan. 2014.

[4] R. Hu, Y. Li, and Y. Zhang, "Adaptive Resource Management in PaaS Platform Using Feedback Control LRU Algorithm," International Conference on Cloud and Service Computing, 2011.

[5] C. S. Pawar, and R. B. Wagh, "Priority Based Dynamic Resource Allocation in Cloud Computing with Modified Waiting Queue," 2013 International Conference on Intelligent Systems and Signal Processing (ISSP), 2013.

[6] Buyya.R.et al., "Market-Oriented Cloud Computing: Vision, Hype and Reality for Delivering it Services as Computing Utilities," c 2008.

[7] Ghalem Belalem, Said Limam, "Fault Tolerant Architecture to Cloud Computing using Adaptive Checkpoint," International Journal of Cloud Applications and Computing, 1(4), pp. 60-69, 2011.

[8] Malte Schwarzkopf, Derek G. Murray, Steven Hand, "The Seven Deadly Sins of Cloud Computing," Research University of Cambridge Computer Laboratory.

[9] Sandeep Sharma, Sarabjit Singh, and Meenakshi Sharma, "Performance Analysis of Load Balancing Algorithms," World Academy of Science, Engineering and Technology, 2008.

[10] R.Buyya et al, "Cloud Computing Principles and Paradigms," Published by John Wiley Sons, Inc., Hoboken, New Jersey.

[11] Ghemawat S, Gobioff H, Leung S T., "The Google file system[C]", ACM SIGOPS Operating Systems Review. ACM, 2003, 37(5): 29-43.

[12] Shvachko K, Kuang H, Radia S, et al., "The hadoop distributed file system[C]," Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on. IEEE, 2010: 1-10.