Deduplicaton In Cloud

Piyush Bhagwatkar , Harsh Pardeshi , Sanket kalokhe, Yogini Pawar
Students Guru Gobind Singh Polytechnic, Nashik , Nashik, Maharashtra, India.
Guide By :Prof. C. R. Ghuge

***Abstract*—**on-demand network access to a shared pool of reconfigurable computing resources, including servers, storage, and applications, is made possible by cloud computing. These shared resources can be quickly provided to consumers by allowing them to pay only for what they use. Delivering storage resources to customers via the Internet is known as cloud storage. Private cloud storage is limited to specific organizations, and cloud storage poses higher data security risks. Therefore, private cloud storage is created by leveraging standard machines within an organization, and critical data is stored there. As the use of such private cloud storage increases, connectivity issues, performance and storage requirements, privacy and security, and data integrity increase. Implement offline data storage and synchronization mechanisms in case of connectivity issues. Increasing storage requirements lead to scaling cloud storage by adding storage nodes. Storage nodes in cloud storage must be load-balanced during such expansion. Data must be moved between storage nodes in order to maintain load across many storage nodes. The network bandwidth is used higher during this data movement. The core idea behind it is to develop a deduplication-based dynamic load balancing algorithm to balance storage node load during private cloud storage expansion. To maintain data integrity, security, and privacy, we use the AES and SHA algorithms. SHA algorithm is also used to avoid duplicates.

***Keywords*— Cloud computing, cloud solutions, reliability, load balancing, encryption, secure deduplication, data integrity**

Technical Keywords:
Advanced encryption standards, load balancing, deduplication, secure hash keys, and cloud computing

I. INTRODUCTION

Cloud computing is very important in information technology today. Access to a common pool of reconfigurable computer resources, such as servers, storage, and applications, is made possible via cloud computing. The hosting services provided to the users are done through the Internet. There are disaster risks in the cloud, such as problems with connectivity, performance, privacy and security, and data management. To solve the connection problem, we can implement offline storage and synchronization mechanisms. To improve Performance, load balancing is an important task for performing cloud operations and therefore also for deduplication. Load balancing is required as cloud computing expands and more clients around the world expect more services and better outcomes. Load balancing ensures efficient use of resources for customers on demand and increases overall cloud performance. Each growing volume of backup data in cloud storage can be a significant challenge, so we can use de-duplication mechanism to eliminate duplicate data. In order to distribute client requests among accessible remote nodes, numerous techniques have been devised. This project's main goal is to provide an offline synchronization and storage system, as well as a dynamic load-balancing technique based on duplication to balance the strain on storage nodes as storage grows. Private cloud.

II. MOTIVATION OF PROJECT

The main motivation of the system is to remove the load on cloud-based servers and avoid data duplication by using certain methods and algorithms. This system is basically run on hash code detection techniques used to avoid storing lots of files on cloud servers. For a load-balancing engineering system, split the file into three parts and store it in three different locations, and access is restricted to authorized or authorized persons with only login credentials with a valid user key issued by the administrator. This system has the feature of requesting information from the client for login and sending the username, password and private key to the user with the help of the administrator. They have login information as well as private key so login can easily perform upload, delete and upload operations. Using Advanced Encryption Standard (AES) and Secure Hash (SHA) algorithms, data security and load balancing will be managed. The hash code generates a code according to the file data and is stored in the database if the code is the same then the duplicate file message will come if the code is not unique then the file is split into three different parts and stored in three different locations. If the user tries to delete or download the file without the private key and credentials, it fails. The credentials are matched, then the three blocks are merged into a single file and delete/upload operations are performed, making the process faster and more secure. If the connection is not available, use the offline storage and synchronization mechanism.

III. LITERATURE REVIEW:

Due to rising costs, IT companies began to outsource IT services. IT services are maintained by specialized companies, so-called service providers. This gave rise to cloud computing. Cloud computing is a computing environment in which resources such as computing power, storage, networks, and software are abstracted and offered as services on a distributed network. Cloud computing is a technology that performs tasks by sharing and using existing resources and applications in a distributed network environment. Resources can be easily allocated and released by service providers. Many users demand services in the cloud that behave like the Internet at scale. Due to the exponential growth of users and their needs, various companies are adopting cloud his computing. There are cloud computing data centers around the world to enable cloud computing. Various cloud services such as B. Pay-as-you-go pricing that is offered at a low price without intervention by the owners and administrators of these services.

Cloud computing consists of several characteristics:

•On Demand -Cloud services are provided on demand. The user can complete the task at any time.

• Widespread network access – In cloud computing, resources are distributed throughout the network. These resources are accessed through various mechanisms.

• Resource Pooling - Resources are pooled accordingly. Resources are dynamically allocated and released accordingly

•Scalability - The amount of resources can be increased at any time according to customer needs.

As usual, cloud computing has some problems. These issues come with the high number of requests to service these clouds. Load balancing, redundancy, and fault tolerance are such issues. 444.4 million Service users worldwide consistently submit service requests to the cloud for their storage and compute needs. Cloud computing should provide the abstraction that the user's tasks are executed exclusively and provide error-free output. As demand grows, the resources serving those demands must also be upgraded and upgraded. Cloud computing should work in a way that the load is distributed. This is where a technique called load balancing can help. Cloud load balancing is a method of distributing services and computing resources in a cloud computing environment. Load balancing allows organizations to manage workload demands by allocating resources across multiple computers, networks, or servers in the cloud. By sharing the workload, tasks are executed concurrently. This helps with the basic idea that the entire load should not be shifted to just one server. All servers and resources work together, and finally, when all resources have completed their assignments, output is produced. As cloud technology became more popular, so did data sharing and storage The less data you store, the less hardware resources you need, so the amount of data you need to manage continues to grow. Adding hardware to store more data increases costs, so service providers should consider this. It should also be cheaper for the user than actually storing the data on their side. Data deduplication is he one of the most popular storage technologies today because it can save companies a lot of storage and bandwidth costs for storing data. This is good news for cloud he providers. Less storage means less hardware. If you can deduplicate the data you store, you can make better use of your existing storage space. This allows you to use your existing storage space more efficiently and save money. The less you store, the less you back up. This means less hardware and less backup media. Less data to store means less data to send over the network in the event of a disaster. This means you can save hardware and network costs over time. Data deduplication is truly a game changer and saves you money.

The business benefits of data deduplication are:

- Reduced Backup Cost
- Reduced Hardware Costs
- Reduced Business Continuity and/or Disaster Recovery Costs
- Storage Efficiency
- Network Efficiency

Objects (usually files or blocks) are compared and data sets An object (a copy) that already exists in is deleted. The deduplication process removes non-unique blocks

IV. TECHNOLOGIES AVAILABLE IN THE CLOUD

The following techniques are currently in vogue in the cloud:

• Vector Dot- A. Singh et al. proposed a new load balancing algorithm called Vector Dot. It manages the hierarchical complexity of the data center and the multidimensionality of resource loads across servers, network switches, and storage in a flexible data center that integrates server virtualization and storage. Vector Dot uses the dot product to differentiate nodes based on element requirements and eliminates overhead on servers, switches, and storage nodes.

• CARDBOARD-R. Stanojevic et al. proposed a CARTON mechanism for unified cloud control using LB and DRL. LB (Load Balancing) is used to evenly distribute the work across different servers so that associated costs can be minimized, and DRL (Distributed Rate Limiting) is used to ensure that resources distributed in a way that maintains proper resource allocation. DRL also adapts the server's capacity to dynamic workloads so that the performance levels of all servers are

equal. With very low computation and communication time, this algorithm is very simple and easy to implement.

• Comparison and Equilibrium - Y. Zhao et al. solved the problem of internal cloud load balancing between physical servers by direct adaptive migration of virtual machines. The load balancing model is designed and implemented to reduce the migration time of virtual machines through shared memory, to balance the load between servers based on their CPU or I/O usage, and more. And to maintain virtual machine downtime in the process. A COMPARISON AND BAL-ANCE distributed load balancing algorithm is also proposed, which is based on sampling and achieving balance very quickly. This algorithm ensures that the migration of virtual machines is always from a high-cost physical server to a low-cost server, but assuming that each physical server has enough memory, this is a weak assumption.

• Event-driven - V. Nae et al. presented an event-based load balancing algorithm for real-time massively multiplayer online games (MMOG). This algorithm, after receiving capacity events as input, analyzes its components in the resource context and overall state of the game session, thus generating session load balancing actions. Game. It can span a game session across multiple resources based on different user loads, but sometimes manifests QoS violations.

• Strategic planning for LB of VM resources - J. Hu et al. proposed a scheduling strategy for VM resource load balancing using historical data and the current state of the system. This strategy achieves the best load balancing and reduced movement by using a genetic algorithm. It helps to solve the problem of load imbalance and high migration costs, thus enabling better resource utilization.

• CLBVM- A. Bhadani et al.proposed a centralized load balancing policy for virtual machines (CLBVMs) that balances the load equally in distributed cloud/virtual machine environments. This strategy improves the overall performance of the system but does not consider fault-tolerant systems.

• LBVS-H. Liu et al. proposed a load-balanced Virtual Storage (LBVS) strategy that provides a large-scale network data storage model and a Storage-as-a-Service model based on Cloud Storage. Storage virtualization is achieved using a three-layer architecture and load balancing is achieved using two load balancing modules. It improves concurrency efficiency by using replica balancing, which further reduces response times and improves disaster recovery. This strategy also improves the storage resource utilization rate, flexibility and robustness of the system.

• LB-Y-based task scheduling. Fang et al. discussed a load-balancing two-tier task scheduling mechanism to respond to dynamic user requests and achieve high resource utilization. It achieves load balancing by mapping tasks first to virtual machines and then to virtual machines for resource storage, thus improving task response time, resource usage, and performance. Overall system on the cloud computing environment.

• Foraging behavior of bees - M. Randles et al. studied a bee-based decentralized load balancing technique, a nature-inspired algorithm for self-organizing It performs global load balancing through local server actions. System performance improves as system diversity increases, but throughput does not increase as system size increases. It is most suitable for conditions requiring diverse types of services.

• Biased Random Sampling - M. Randles et al.studied a distributed and scalable load balancing approach that uses system domain random sampling to achieve self-organization, thus balancing the load on all nodes in the system. System. System performance is enhanced with high resources and the like, resulting in increased throughput through increased efficient use of system resources. It declined along with the increase in population diversity.

• Active Clustering - M. Randles et al. studied self-aggregation load balancing technique, which is a self-aggregation algorithm to optimize task assignment by connecting similar services using local rewind. System performance is improved with high resources, which increases throughput by using those resources efficiently. It declines with increasing diversity of the system.

• ACCLB-Z. Zhang et a proposed a load balancing mechanism based on the theory of ant colonies and complex networks (ACCLB) in an open cloud computing federation. It uses the small-world and zero-scale characteristics of a complex network to achieve better load balancing. This technique overcomes heterogeneity, adapts to dynamic environments, has excellent fault tolerance, and has good scalability, which improves system performance.

• (OLB + LBMM) - S.-C. Wang et al. proposed a two-stage scheduling algorithm that combines OLB (Opportunity Load Balancing) and LBMM (Min-Min Load Balancing) scheduling algorithms to take advantage of better execution efficiency and maintain balance. System load. The OLB scheduling algorithm keeps each node in a working state to achieve the load balancing goal, and the LBMM scheduling algorithm is used to minimize the execution time of each task on the node, thereby minimizing the execution time. Overall implementation time. Thus, this combined approach contributes to the efficient use of resources and improved work efficiency.

• Perception of Decentralized Content - H. Mehta et al. proposed a new content-aware load balancing policy called the Workload and Client-Aware Policy (WCAP). It uses a parameter named USP to specify unique and special properties

of queries and compute nodes. The USP helps the scheduler decide which node is best suited to handle requests.

• This strategy is implemented in a decentralized way with low costBy using content information to refine the search, it improves the overall search performance of the system. It also helps to reduce the idle time of computational nodes, thus improving their utilization.

• Server-based LB for Distributed Internet Services - A. M. Nakai et al. proposed a new server-based load balancing policy for globally distributed web servers. It reduces service response times by using a protocol that limits the redirection of requests to the nearest remote servers without overloading them. A middleware is described to implement this protocol. It also uses heuristics to help the web server withstand overload.

• Join-Idle-Queue-Y. Lua et al. proposed the Join-Idle-Queue load balancing algorithm for dynamically scalable web services. This algorithm provides large-scale load balancing with distributed dispatchers by first balancing the inactive processors among the dispatchers for the availability of non-operating processors. Works on each dispatcher, then assigns tasks to the processors to reduce the average queue length per processor. By removing the load balancing work from critical request processing, it effectively reduces the system load, does not cause any communication overhead when the job arrives and does not increase the time actual feedback.

V.COMPARISON OF EXISTING AND CURRENT SYSTEM

Many researchers have worked in this section, and we discuss previous work related to load balancing in the cloud using various techniques. Current cloud server storage technology makes updating, deleting, and downloading files less secure. Fewer load balancing techniques and no deduplication. The current system only provides space on the server and does not avoid file duplication. This project uses hash codes for file contents. If this code is found in the database, the system will generate a duplicate file her message for the user. Otherwise the file is split into his 3 parts and stored in 3 different locations, thus splitting the load and load balancing. It's done automatically. It also uses AES algorithm for encryption and secret key mechanism to ensure privacy and security. Offline storage and synchronization mechanism to resolve connectivity unavailability issues.

VI.OBJECTIVES AND GOALS

1. The major goal of this system is to keep the cloud loaded.
2. The system uses the AES encryption method and a private key mechanism to preserve security and privacy.
3. To avoid multiple storage of duplicate files on the Cloud, this system primarily uses hash code detection algorithms.

VII. SCOPE STATEMENTS

1. This system has the feature of requesting information from the client for login and sending the username, password and private key to the user with the help of the administrator
2. They have login information as well as private key for login which can easily perform upload, delete and upload operations.
3. Using the Advanced Encryption Standard (AES) and Secure Hash (SHA) algorithm, data security and load balancing will be managed.
4. The hash is generated based on the file data and stored in the database if the code is the same then the duplicate file message will come if the code is not unique then the file is split into three different parts and stored in three different locations.
5. If the user tries to delete or download the file without the private key and credentials, it fails.
6. The credentials are matched, then the three blocks are merged into a single file and delete/upload operations are performed, making the process faster and more secure.

VIII. MAJOR RESTRICTIONS

1) The user must enter the private key that was sent to the registered email address.
2) One encrypted file will be submitted as a result.
3) The file's hash code.
4) The file will be downloaded after being decrypted.

IX. APPLICATION

1. Allocate resources among several computers or networks to enable organizations to manage application or workload demands.
2. Provide a single internet service from a number of servers; this is sometimes referred to as a server farm.

X. ALGORITHM/METHOD DETAILS

Advanced Encryption Standard (AES): Encrypts data using the AES algorithm. AES includes three block ciphers: AES-128,

AES-192, and AES-256. Each cipher encrypts and decrypts data in 128-bit blocks using 128-, 192-, and 256-bit encryption keys, respectively. (Rijndael was designed to handle additional block sizes and key lengths, but the functionality was not carried over to AES.) Symmetric (secret key) ciphers are it uses the same key, so both sender and receiver must know it. use the same private key.All key lengths are deemed sufficient to protect sensitive information down to the "secret" level using "top secret" information requiring key lengths of 192 bits or 256 bits. Considered. There are 10 rounds of final ciphertext output for 128-bit keys, 12 rounds for 192-bit keys, and 14 rounds for 256-bit keys.

Secure Hash Algorithm (SHA): The SHA algorithm generates a hash code based on the file contents. A cryptographic hash function is a mathematical operation performed on digital data. Data integrity can be determined by comparing the computed "hash" (the result of running an algorithm) against a known and expected hash value. For example, computing a hash of a downloaded file and comparing the result to previously published hash results can reveal whether the download has been altered or tampered with. An important aspect of cryptographic hash functions is their collision resistance. You cannot have two different input values that result in the same hash output.

INPUT

1. Enter your login information.
2. The uploading of a file.
3. Download and delete requests with a secret pin.

EXPECTED OUTPUT

1. A file that is encrypted will be uploaded.
2. The file's hash code.
3. The file will be downloaded after being decrypted.

CONCLUSION

These systems offer the architectural design to prevent cloud disasters like issues with connection, performance, privacy & security, and data management. We are leveraging mechanisms like load balancing, de-duplication, offline storage & sync, and AES encryption to solve these issues.

REFERENCES

[1.] J. Wu, L. Ping, X. Ge, Y. Wang, J. Fu, Cloud Storage as Infrastructure for Cloud Computing, in Proc. 2010 International Conf. Intelligent Computing. Cognitive information. (ICICCI), Kuala Lumpur, 2010, pp. 380-383.

[2.] Reinsel, The Decade of the Digital Universe - Are You Ready, IDC White Paper, http://www.emc.com/collateral/analyst-reports/idc-digitaluniverse-are-you-ready.pdf, 2010.

[3.] P. Xie, Research on deduplication technology for storage systems, computing. Sci., Vol.41, No. 1. S. 22.-30. January 2014.

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

[5.] C. S. Pawar und 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.] GhalemBelalem, Said Limam, Fault Tolerant Architectures for Cloud Computing Using Adaptive Checkpoints, International Journal of Cloud Applications and Computing, 1(4) , S. 60 - 69, 2011.

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

[9.] Sandeepsharma, Sarabjitsingh and Meenakshi Sharma, Performance Analysis of Load Balancing Algorithms, World Academy of Sciences

[10.] Engineering and Technology, 2008.