LOAD BALANCING IN CLOUD COMPUTING: A REVIEW

[1]M.R.BanuPriya, [2] Dr. D Francis Xavier Christopher
[1]Associate Professor, [2]Director,
[1]Department of Computer Applications,
[1]Kongunadu Arts and Science College, Coimbatore, India

Abstract : Cloud computing is an emerging technology for storing data over the internet. It's a hybrid technology which refers to manipulating, configuring and accessing the applications online. Load balancing is to be able to distribute the incoming requests over a number of backend servers in the cluster. Load balancing helps to make networks more efficient and aim at achieving the high performance, user satisfaction, minimizing the response time of the task and also to improve the resource utilization. It is a key component of highly-available infrastructures commonly used to improve the performance and reliability of websites, applications, databases and other services by distributing the workload across multiple servers. It can be performed by an application by either physical or virtual that identifies in real time which server in a pool of servers can meet a given client request, while ensuring heavy network traffic does not unduly overwhelm a single server. This paper presents a review of the existing techniques for load balancing.

***IndexTerms*-Cloud computing, Load balancing, Clustering, Resource clustering, Resource utilization**

I. INTRODUCTION

The cloud is just a metaphor for the Internet. Cloud Computing is a predominant technology which uses the internet and central remote servers to maintain data and applications. It helps end users and businesses to use applications without installation and allowed to access their personal files at any computer with internet access. Cloud computing is an on-demand availability of computer system resources mainly data storage and computing power, without direct active management by the user. Cloud load balancing includes workload traffic and demands that exist over the internet. Hence, the workload on the server growing so fast which leads to the overloading of servers mainly for popular web server.

II. CLOUD COMPUTING

"Cloud" the technology of distributed data processing in which some scalable information resources and capacities are provided as a service to multiple external customers through Internet technology[13]. The following are the types of services rendered that are commonly referred to as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) which is called as cloud computing stack because they build on top of one another. Almost in all areas cloud computing made many things easier like interoperability, secure storage, 24×7 uptime, etc. This advancement leads to many drastic changes in a proper way of treating information. The same rationale applies to operating systems, middleware or platform software, and application software. IT resources are made available to be shared by numerous cloud consumers which results in increased or even maximum possible utilization of resources. Operational costs and inefficiencies can be further reduced by applying proven practices and patterns for optimizing cloud architectures, their management and governance. As with all cloud computing services, it provides access to computing resource in a virtualized environment, "The Cloud", across a public connection, usually the internet. With IaaS, however, the client is given access to virtualized components in order to build their own IT platforms. Abstraction of the infrastructure so applications are not locked into devices or locations and can be easily moved if needed.

III. LOAD BALANCING

Efficiently distributing incoming network traffic across a cluster of backend servers, also known as a server farm or server pool. Load balancing is a procedure used to distribute workloads uniformly across various servers or other compute resources to optimize network efficiency, reliability and capacity. Load balancing is achieved by an appliance - either physical or virtual – which identifies in real time which of the server in a pool can best to meet a given client request, while ensuring heavy network traffic doesn't unduly overwhelm a single server

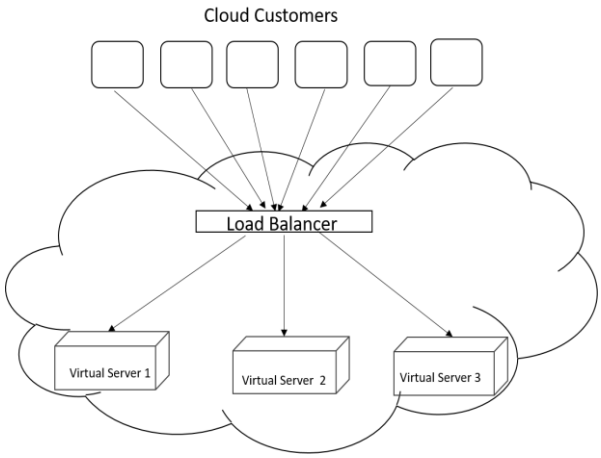

Fig: 1 Load balancer in Cloud Environment

3.1 Why are Load Balancers needed?

Volumes of traffic are increasing and applications are becoming more complex. Load balancers provide the substratum for building flexible networks that meet evolving demands by improving performance and security for many types of traffic and services, including applications. Load balancers will choose the server to forward the traffic based upon two factors.
Health of the Server and Predefined Conditions

3.1.1Health of the Server

This can be found by using the continuous test which checks whether the system can deliver the response by sending the request to it from load balancers. If there is no response then the load balancer will switch over to another server.

3.1.2Pre-defined Conditions

This can be termed as the algorithm which use different conditions to select a server. By maximizing network capacity and performance, load balancing provides failover. If one server fails, a load balancer must immediately redirects its workloads to a backup server. In contrast, software load balancing runs on virtual machines (VMs) or white box servers, most probably as a function of an application delivery controller (ADC). ADCs typically offer additional enabling users to automatically scale up or down to mirror traffic spikes or decreased network activity.

3.2 Metrics for Load Balancing

Some of the common metrics which are followed are:

3.2.1 Throughput

It is used to calculate all tasks assigned whose execution has been completed. The performance of any system can be improved if throughput is high.

3.2.2 Fault Tolerance

It means recovery from failure. The load balancing treated be a good fault tolerant technique.

3.2.3 Migration time

The time to migrate the job resources from one node too the other nodes. It must be minimized in order to enrich the performance of the system.

3.2.4 Response Time

It is the amount of time that is taken by a particular load balancing algorithm to response at a skin a system. For better performance parameter should be minimized for a system.

3.2.5 Scalability

It is the capability of an algorithm to perform Load balancing for any predetermined number of nodes of a system.

3.3 Server Load balancing

Server farms achieve high scalability and high availability through server load balancing, a technique that makes the server farm appear to clients as a single server. Load balancing server distributes service requests across a group of real servers and thus makes those servers look like a single big server to their respective clients. Real servers are behind a URL that implements a single virtual services. The selected server is to provide the load balancing algorithm with the required input data, the load balancer also retrieves information about the servers' health and load to verify that they can respond to traffic.

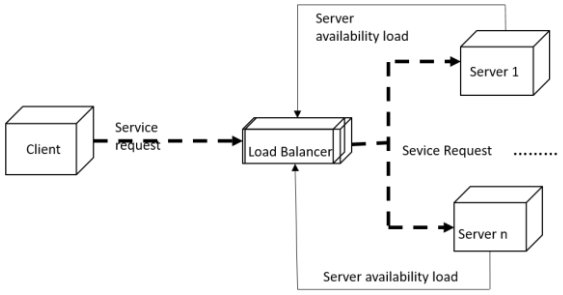

Fig 2: Illustrating load balancer architecture.

The load balancer architecture illustrated in Figure 1 is just one of several approaches to decide which load balancing solution is the best for the infrastructure, need to consider ***availability*** and ***scalability*.**

Availability is defined by *Uptime* -- the time between failures. *Downtime*—is used to detect the failure of the node, to be repair it, perform required recovery, and restart tasks again. At uptime the system must respond to each request within a predetermined, well-defined time. If this time is exceeded, the client understands this as a server malfunction. Basically, availability is redundancy in the system: If one server fails, the others take over the failed server's load transparently. The disappointment of an individual server is invisible to the client.

Scalability means that the system can assist a single client, as well as thousands of simultaneous clients, by meeting quality-of-service requirements such as response time. At an increased load, a high accessible system can maximize the throughput almost linearly in proportion to the power of added hardware resources. In the above architecture high scalability is reached by distributing the incoming request over the servers. If the load increases, additional servers can be added, as long as it does not become the bottleneck. In order to reach high availability, the load balancer must monitor the servers to avoid forwarding requests to overloaded or dead servers. Furthermore, the load balancer itself must be redundant too.

3.4 Load balancing Server Techniques

In general, server load balancing solutions are of two main types:

***Transport-level* load balancing** -- such as the DNS-based approach or TCP/IP-level load balancing -- acts independently of the application payload.

***Application-level* load balancing** - uses the application payload to make load balancing decisions.

Load balancing solutions can be further classified into:

Hardware-based load balancers- It is a specialized set of hardware boxes that include application-specific integrated circuits (ASICs) which is customized for a particular use. ASICs enables the high-speed forwarding of network traffic without an overhead of operating system. Hardware-based load balancers are frequently used for transport-level load balancing. In general, hardware-based load balancers are much faster than software-based solutions. Their drawback is their cost.

In contrast, **software-based load balancers** run on standard operating systems and standard hardware components such as PCs. For Internet services, a server-side load balancer is usually a software program that is attending on the port where external clients connect to access services. The load balancer forwards the incoming requests to one of the "backend" servers, which usually replies to the load balancer. This helps the load balancer to reply to the client without knowing the client about the internal separation of functions. It also prevents clients from contacting back-end servers directly, which may have security benefits by hiding the structure of the internal network and preventing attacks on the kernel's network stack or unrelated services running on other ports.

3.5 Load balancers Features

Hardware and software load balancers may have many special features. The important feature of a load balancer is to be able to distribute incoming requests over a number of backend servers in the cluster according to a scheduling algorithm.

A load balancer performs the following functions:

- Distributes client requests or network load efficiently across multiple servers.
- Ensures high availability and reliability by sending requests only to servers that are online
- Provides the flexibility to add or subtract servers as demand dictates.

3.6 Load Balancing Algorithms And Methods

Load balancing uses various algorithms, called load balancing methods, which is used to define the criteria that the ADC (Application Delivery Controller) appliance used to select the service in which to redirect each client request. Different load balancing algorithms use different conditions.

3.6.1 Asymmetric load

A ratio can be manually assigned to some backend servers to get a greater share of the workload than other servers. This is sometimes used as a rough way to account for some servers having more capacity than the other and may not always work as preferred.

3.6.2 Priority activation

If the number of available servers drops below a certain number, or load gets too high, backup servers can be brought online.In these algorithms the system state is not considered and also in correlative and unchangeable environments.

3.7 Classification Of Algorithms

The following are the algorithms which are frequently reviewed and considered by the authors for their research.

3.7.1 Static Algorithms

In these algorithms the system state is not considered and also in correlative and unchangeable environments.

- ***The Least Connection Method***-When a virtual server is configured to use the least connection, it chooses the service with the smallest number of active connections.
- ***The Least Response Time Method***-This method selects the incoming service with few active connections and the lowest average response time.
- ***The Least Bandwidth Method*** -This method selects the service that is currently serving the least amount of traffic, measured in megabits per second (Mbps).
- ***The Least Packets Method*** - This method selects the service that has expected the fewest packets over an indicated period of time.
- ***The Custom Load Method***-This method helps the load balancing appliance chooses a service that is not handling any active transactions.
- ***Round Robin Algorithm***-This method continuously rotates a list of services which are attached to it. When the virtual server receives a request, it assigns the connection to the first service in the list, and further moves that service to the bottom of the list. Round Robin is definitely the most widely used algorithm. It's easy to implement and easy to understand.[9]Would be more effective if the equipment that we are load balancing is roughly equal in processing speed, connection speed, memory resources.
- ***Weighted Round Robin***-The Weighted Round Robin is similar to the Round Robin in a sense that the method by which requests are assigned to the nodes is still recurrent, although with a twist. The node with the higher specs will be apportioned a greater number of requests.
- ***Least Connections***-This algorithm takes into consideration the number of current connections each server has. When a client attempts to connect, the load balancer will try to govern which server has the least number of connections and then assign the new connection to that server.

- ***Weighted Least Connection***-Weighted Least Connections algorithm does the same as Least Connections what Weighted Round Robin does to Round Robin. That is, it introduces a "weight" component based on the respective capacities of each server. Just like in the Weighted Round Robin, you'll have to specify each server's "weight" beforehand.
- ***Random***-As its name implies, this algorithm matches clients and servers by random, i.e. using an underlying random number generator. [3]In cases wherein the load balancer receives a large number of requests, a Random algorithm will be able to distribute the requests evenly to the nodes. So like Round Robin, the Random algorithm is sufficient for clusters consisting of nodes with similar configurations (CPU, RAM, etc.).

3.7.2 Dynamic Algorithms

Some of the Dynamic algorithms are proposed by the researchers.

Ant Colony Optimization with Particle Swarm (ACOPS): [1] proposed an algorithm which helps in reducing the amount of computing time that is used in scheduling. Here, the concept of hybrid heuristic method is proposed which combines the two algorithm ACO and PSO. The proposed algorithm helps to serve the VM requests. It helps in reducing the schedule time and also it increases the speed of the scheduling procedure. ACOPS also produced random dynamic request streams for continuous scheduling. Also it make use of the historical requests workload for predicting the new input request. So, that the input can be executed with a limited information.

Ant Colony Optimization (ACO) and Honey Bee Algorithm (BA): In [2] the author proposed a detailed description of ACO which is based on the principles of natural system behavior. This provides the technique of finding the optimal path between ants nest and the food source. Communications can be done through pheromone mechanism, a chemical signal which guides the other ants to reach the food source. The quality of the pheromone helps the ants to choose the shortest path.
Also the detailed description of BA, which is based on the behavior of honey bee colonies. Single colony bees contains a queen bee. Functions such as rearing, maintaining the hive and collecting the ne
ctar. Two types of bees. Scout and Forager bees. Scouts are workers and find the food source by waggle dance in front of the hive and deposits the nectar. Forager will follow the scouts and may waggle dance for the highly profitable food. Dances can be round and waggle. Round instructs the foragers to learn the approximate distance from hive to the food site. Waggle is to advertise about the quality, quantity, direction of food and distance.Also the research travels based on OS which contains HAProxy load balancer. It compares with round robin, static Round-Robin and Least connection. As a result, round robin algorithm [9] is considered to be the best when there is a comparison of resource utilization, network connection and network request. This analysis is based on the parameters such as nature, throughput, resource utilization, number of connection per second, number of request per second.

Cluster-Based Algorithm: In the paper [3] the author proposed decentralized heterogeneous network for load balancing. This algorithm presents the features like heterogeneity, scalability, low network congestion and managing the bottle neck node. Clustering concept is followed. Master-Slave architecture. ICC is available in every cluster. Each slave is connected with one master. Master maintains a table for load distribution and it is updated frequently. When a task is assigned to each slave completion of each task is reported to the master. Two parts of algorithm is considered. Load distribution among masters and load distribution from master to slave. It describes the ability of master node based on incoming task. This algorithm states that one node can belong to only one cluster. To evaluate its effectiveness it can also belong to more than one cluster.

In paper [4] the author has proposed an algorithm which works well in heterogeneous nodes environment. It makes use of K-Means clustering approach to divide the virtual machines in to cluster. For each cluster, a list is maintained about the VM's. It helps to reduce the overhead of scanning. This algorithm helps in obtaining better results in terms of waiting time, execution time,turn around time and throughput. Clusters has been chosen to the highest prime factor. During the assigning of task, among the cluster the load balancer will match the suitable VM. Some of the parameters are considered such as response time or waiting time, execution time, turnaround time, throughput

Load Balancing- Resource Clustering(LB-RC): The author[5] proposed an algorithm LB-RC helps in finding the optimal set of serves for assigning the task in order to balance the load of the servers in a long-term process. Resource clustering, merging of clusters, resource optimization, task assignment policy were proposed. In resource clustering ,servers are grouped based on their loads in to homogeneous clusters. Merging of clusters helps to merge the null and the small clusters. In resource optimization, Bat algorithm is used for optimum problems and finds the optimal use of clusters for detecting multiple objects. Finally task deployment is done by using best fit VM instances to the optimal servers. Task assignment policy helps to minimize the makespan and execution cost.

Task Based Load Balancing(TB-LB): In paper[6] the author proposed an algorithm task based approach towards load balancing. TB-LB combines heuristic algorithms such as genetic, simulated healing, particle swarm optimization. It helps to cluster the VM's in to groups with K-Means approach. As per the incoming tasks, the load scheduler selects the cluster and assign the task. The load balancing selects the available VM and the job get assigned. Using this algorithm, dividing and allocating the available VM's to K number of groups. Cloud load balancer maintains a list of VM's in a cluster. The algorithm uses the concept of assigning priority to it and makespan get improved. K-means clustering reduces the time required to search suitable VM and also it enhances the performance of the system.

Fuzzy C Means Cluster Based Load Balancing (FCM-LB): In paper [7] the author proposed an algorithm using MAT-LAB. It works well in heterogeneous node environment. It reduces scanning overhead by the division of virtual machines in to clusters. This algorithm aims to improve the throughput, execution time, response time and turnaround time. This can be done with the

concept of Fuzzy C-means clustering approach for the purpose of clusters. Load balancer will gather the details of all clusters and maintain as a list. Also, it maintains the list of VM's of each cluster. Thus helps in reduces the overhead of scanning process.

In paper [8] the author compared approaches which is a combination of Honeybee Foraging Algorithm, Active Clustering Algorithm and Ant Colony Optimization, which helps for load rebalancing on cloud. Honeybee foraging helps to improve an average execution and waiting time of tasks. Active clustering is used to increase the throughput and utilization of resources.

The author in the paper [9] proposed an approach for enhancing the load distribution process. Service broker policy is a concept by which the proposed algorithm decides to distribute the arrived task among data center. Another concept is optimize response time by choosing according to their response time. Round Robin is also used for the concept of distribution.

Grammatical Evolution Enhanced Simulated Annealing (GE-SA): The algorithm in the paper [10] mainly deals with two key elements. Cooling schedule parameter and neighborhood structure. First element deals with the temperature where it should not be too slow or too quick which affect the performance. Second new solutions are generated by a predefined procedure.

In the paper [11], the author proposed an algorithm which mainly achieve a better utilization of CPU. This can be achieved with the approach of Ant Colonies. Since ACO provides the best results for Travelling Salesman Problem for load balancing. This algorithm provides a feasible solution for load balancing based on ACO and Ant Colony System which is done with reformulation. In which the distance between nodes and every node is connected with all nodes except the node which is already visited of the datacenter.

CONCLUSION

Load balancing techniques helps in proper utilization of resources and improve the performance of system. Load balancing is a foremost topic to be discussed in cloud computing. The paper also examined some existing load balancing algorithm proposed by various authors and their views which provide better scheduling and resource allocation techniques. But still there is need of improvement in the strategy of resource allocation and clustering algorithms.

REFERENCES

[1] K.-M., Tsai, P.-W., Tsai, C.-W., & Yang, C.-S. (2014). A hybrid meta-heuristic algorithm for VM scheduling with load balancing in cloud computing. Neural Computing and Applications, 26(6), 1297–1309. doi: 10.1007/s00521-014-1804-9

[2] Mbarek, F., & Mosorov, V. (2018). Load balancing algorithms in heterogeneous web cluster. 2018 International Interdisciplinary PhD Workshop (IIPhDW). doi:10.1109/iiphdw.2018.8388358

[3] Dhurandher, S. K., Obaidat, M. S., Woungang, I., Agarwal, P., Gupta, A., & Gupta, P. (2014). A cluster-based load balancing algorithm in cloud computing. 2014 IEEE International Conference on Communications (ICC). doi:10.1109/icc.2014.6883768

[4] Kapoor, S., & Dabas, C. (2015). Cluster based load balancing in cloud computing. 2015 Eighth International Conference on Contemporary Computing (IC3). doi:10.1109/ic3.2015.7346656

[5] Adhikari, M., Nandy, S., & Amgoth, T. (2019). Meta heuristic-based task deployment mechanism for load balancing in IaaS cloud. Journal of Network and Computer Applications, 128, 64–77. doi:10.1016/j.jnca.2018.12.010

[6] Sajjan, R. S., & Biradar, R. Y. (2018). Load Balancing using Cluster and Heuristic Algorithms in Cloud Domain. Indian Journal of Science and Technology, 11(15), 1–7.doi:10.17485/ijst/2018/v11i15/118729

[7] Geetha Megharaj, Dr.Mohan G.Kabadi, Rajani, Deepa M.(2018).FCM-LB: Fuzzy C Means Cluster Based Load Balancing in Cloud. International Journal of Innovative Research in Science, Engineering and Technology, Volume 7, Special Issue 6, May 2018. (RTCSIT'18)

[8] Ruchika Aggarwal et al, International Journal of Computer Science and Mobile Computing, Vol.6 Issue.6, June-2017, pg. 180-186

[9] Tailong et al., International Journal of Advanced Research in Computer Science and Software Engineering 6(5), May-2016, pp. 552-557

[10] Nasser R. Sabar, Andy Song: Grammatical Evolution Enhancing Simulated Annealing for the Load Balancing Problem in Cloud Computing. GECCO 2016: 997-1003

[11] Moussaddaq, C., Ezzati, A., & Elharti, R. (2017). A new reformulation of the load balancing problem in cloud computing based on TSP and ACO. Proceedings of the 2nd International Conference on Big Data, Cloud and Applications - BDCA'17. doi:10.1145/3090354.3090409