Load Balancing in Cloud Computing

Md Al Yeasin
Department of Computer Science and Engineering,
State University of Bangladesh
Dhaka, Bangladesh

Abstract: The innovation of technology is rapidly arising in every corner of industry, manufacturing fields, corporations and so on. Therefore, the demand of computational analysis is skyrocketing, simultaneoussly the cloud-based computing is facilitating to create more opportunities, thus in order to make it happen, one of the key components which is load balancing that enables us to successfully accomplish our ultimate desire.

Keywords: Cloud Computing, Load Balancing, Data Science, Cloud Architecture, Cloud Components

Introduction: Cloud is dominating beyond the technical industry, it is cost efficient, brings maximum output in an on-demand action. Before a company had to have sufficient budget to setup the business infrastructure from the minute components to the largest one, however, now cloud came as a game changer, a company no longer has to bother about data, its storage, accessibility, security and ease distribution. Every software or platform or infrastructure based cloud now providing all the possible opportunities to get access data, its full security and well distribution within minimum price. As a result, an individual can focus on the core purpose of business or his/her goals in order to attain the uttermost production. Therefore, people are relying on cloud progressively day by day. And to make the cloud work properly, here comes the load balancing that's sole purpose is to make the data distribution well organized as well as in an efficient manner so that cloud can operate from anywhere in the world as an on-demand service.

Literature Review: We can stress out Load Balancer as Static or Dynamic. In the dynamic version, there are three strategies that we get which are Information Strategy, Transfer Strategy and Location Strategy [1]. An extended version of Dynamic Load Balancing is selective borrowing scheme in which a cell borrows channels from the neighboring cells [4]. The parameters that can cause the whole process in Load Balancing are Process Migration, Resource Utilization, Centralized or Decentralized, Forecasting Accuracy, Fault Tolerance, Stability, Overload Rejection, Co-operative and The Natures of Load Balancing [3]. The services like Infrastructure-as-a-Service (IaaS), Platform as-a-Service (PaaS) and Software-as-a-Service (SaaS) [2][5], also the components of Load Balancer and the Load

Balancer as a whole itself availing to bring forth all possible opportunities and help grow the economy of countries. Furthermore, reliability on cloud-based system already has been enriched to the peak by multiple times than ever before. The economy and GDP are mostly linked in between [5][6].

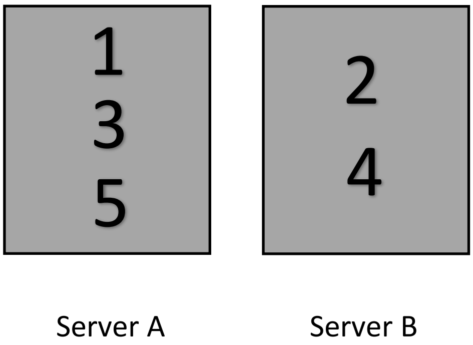
Load Balancer in Practical Manner: As we can see that the distributed computing system is now evolving and companies are trying to provide the best services possible to the customers in order to gain more profit, they are relying on cloud and all kind of online services. By this way they can reach to a customer who is awaiting across the world straightaway, by yielding typical services, companies are consequently gaining their customers faith as well as a reliable solution for them. Additionally, companies can save lots of unnecessary expenses too, ultimately this process is booming the businesses and helping make more profit. As a whole, because of all this demands on cloud system, the main challenge is to maintain it in a systematic order, so that companies do not lose any customer and their faith as well.

To serve the huge amount of data from the server at a single time, it needs a well distribution of requests among servers, so that customers will not have to fall into long queue and finally cancel the request that they want. Here in that case, we use *Load Balancer* which allows us to collect the requests and distribute them among servers then gives response accordingly. However, there are numerous algorithms that we will analyze one by one and at the end, we will try to figure out the best possible efficient way for *Loud Balancer* to execute its tasks successfully.

Key Components of Load Balancer: There are many factors that we have to consider to get the whole insight of *Load Balancer* depending on various types of algorithms and some other cases. **i. Weight of Server** if we consider the weight of a server we can orderly distribute the loads among servers and by this way we can achieve an efficient output. **ii. Number of Requests** it is another factor that is considerable to balance the load. If one server has a smaller number of requests than others then we can assume that other server will successfully take the requests and serve data. **iii. Response Time** it is quite a significant factor because depending on that many requests await, also the distribution process might change due to that reason. So, the response time of a server is a big issue. **iv. Bandwidth** another issue is the speed of internet. Sometimes when a user has trouble in his/her internet, s/he has to wait to send the requests, and because of that s/he might fall behind many requests and might get served late, similarly, if the server has low bandwidth rate, it will bring the same problem, as a result server will not be able to receive requests as well as respond to it in timely manner. **v. Fault Tolerance** every technical component might face difficulties but when there is huge number requests are awaiting, this issue must be take serious into that account and have a probable solution in advance in order to solve it immediately as well as if there comes a problem what will be the alternatives or how it can be mitigated under that circumstances, these should be considered well.

Algorithm Analysis: Now we will see the various types of algorithms which will help know the exact pathway how we can make an effective *Load Balancer*.

Round Robin: By the apply of Round Robin algorithm, we can distribute requests orderly among the servers. For instance, if there are A and B, 2 servers available and 5 requests are incoming, then the 1st one will go to the A, then 2nd one will go to the B, similarly 3rd and 4th will go to the A and B respectively, and finally the 5th will enter into the sever A.



Weighted Round Robin: Unlike to the Round Robin, here the most weighted server will serve first the then the next one, with the descending order it will go on till the least weighted server.

Least Connection: Here the number of requests determines which server has enough to serve and which has not. In that manner, oppositely alike to the Weighted Round Robin, the least number connected server will serve first then this way it will go on.

Weighted Least Connection: In that case, the weight of the server and the number of connections in the server are considered to be executed. The server that has highest weight and lowest number of connections will get the priority to serve the upcoming request.

Chained Failure: This is one of the traditional ways of serving requests, if there are 2 servers, A and B, and each one has capacity of serving 10 requests at a time, then when there will be 12 requests at the same time, at that moment, server A will be fully loaded whereas server will serve only 2 requests. However, this is not an efficient manner to serve data among customers because here we can clearly emphasize that server A will have to take all the loads which might cause delays while on the other hand server B is fully available but not serving equally in order to efficiently distribute data as a whole.

Weighted Response Time: Depending on which server is taking long execution time and which is taking less, we can easily distribute the number of requests based on this property. Because, when a server is

Doing lengthy tasks, then the request will be forwarded to another one which is free or takes less time. In that order, we distribute the requests well and reduce time for the whole process.

Agent Based Adaption Load: In the Load Balancer there will be an agent that will calculate all the parameters like which server has less connection, which one is taking time and which one is ready to serve, the agent will take all of these into account to decide which server will execute the following request.

Consistent Hashing: Each request has a unique id and by hashing that id corresponding to the number of servers, we can get a specific value to assign that request to a specific server and this method is called Consistent Hashing. If there are 4 servers and 2 request id which are 10 and 15, then the hash values we will get are 10%4 = 2 and 15%4 = 3, so the 1st request will go to the server number 2 and 2nd request will go to the server number 3. By applying this hashing method, we can create an efficient Load Balancer which will serve the data in an effective manner.

Advantages of Load Balancer: There are number of advantages of Load Balancer that are discussed below.

Time: With the use of Load Balancer, now it is possible to equally serve huge amount of data simultaneously throughout the world. Many tremendously time consuming or expensive tasks are now can be done by the well distribution of services. Without that, we could not come this far when we can now request at any time to know about weather news or to get any kind of online services from any online platforms. It is bringing ample of opportunities in business and human welfare. Now no one has to wait to get any service at all. On-demand services are everywhere, whenever we wish, we can get access from anywhere in the world which allows us to learn new things without having any boundaries. The most important issue is about time, because of well distribution, these days no clients have to wait for long time to get response. Ultimately, it reduces all potential delays and saves times which is the great advantage.

Scalability: As the number of requests increases by time and it has to work efficiently, so overall, the Load Balancer is quite scalable according to its demand.

High Performance Application: Because of the availability of huge opportunities, people are making high performance application and serve it to the customers easily by the use of Load Balancer.

Recommendation: When the number of requests will reach its limit in the server, at that moment we will have to add more server in order to serve responses. However, if we use Consistent Hashing and later on if we add more servers then all the specific ids or requests will change overall and shuffle throughout the server, in that manner, it will fully change the queue which is not time efficient, an individual will have to request and wait again in order to get data, on the other hand, if we use Cache Memory inside the servers, then we can easily get rid of delays and reduce the overall execution time. Additionally, because of having cache memory, users will not have to load the data again and again, they can simply hit the server and readily get the data because their data already are stored in the cache memory. So, this is the most efficient manner to get the maximum output with minimum time overall. Furthermore, the less the serving time, the more the Load Balancer can serve data and get new requests.

Conclusion: Load Balancer is one of the major components in Cloud Computing. However, more the technology is advancing and new ideas are emerging, the pressure of cloud is getting higher simultaneously, it is mainly because everything is now on cloud-based. As a result, by considering the high demand of cloud in future, we have to figure out more efficient way to distribute our cloud-based operations as it can take the workload and serve the required data throughout the world instantly.

References:

[1] Ali M. Alakeel "**A Guide to Dynamic Load Balancing in Distributed Computer Systems**" IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.6, June 2010

[2] Shivaji P. Mirashe, Dr. N.V. Kalyankar "**Cloud Computing**" JOURNAL OF COMPUTING, VOLUME 2, ISSUE 3, MARCH 2010

[3] Sandeep Sharma, Sarabjit Singh, and Meenakshi Sharma "**Performance Analysis of Load Balancing Algorithms**", World Academy of Science, Engineering and Technology 38 2008

[4] Mani and Tze H Lai "**Characteristics of load balancing and channel assignments in mobile communication systems**". Monash University, Victoria 3800, Australia

[5] Stephen Ezell and Bret Swanson "**How Cloud Computing Enables Modern Manufacturing**", 22 June 2017

[6] Sharyn O'Halloran, Sameer Maskey, Geraldine McAllister, David K. Park and Kaiping Chen "**Data Science and Political Economy: Application to Financial Regulatory Structure**", November 1st, 2016.