An Investigation of Various Cloud Load Balancing Techniques

Akanksha Sen[1], Mr. Jitendra Kumar Tyagi[2]

Abstract- *CLOUD computing is a modern computing platform that is built on virtualization, parallel and distributed computing, utility computing, and service-oriented architecture. Cloud computing has emerged as one of the most influential paradigms in the IT field in recent years, attracting considerable attention from both academia and business. The virtualization theory underpins cloud computing. Both user requests are managed by a cloud-based set of virtual machines. The total efficiency of small servers accessible at the data center decreases as the request sent exceeds the data center's capacity.*

In such cases, load balancing is used to improve data center performance. Load balancing is a technique for transferring loads between various entities, such as CPUs, disk drives, servers, or other types of computers. The primary goal of load balancing is to greatly improve energy efficiency.

Keywords- Cloud Computing, Load balancing, Round Robin, FCFS, Active Monitoring, Throttled.

I. INTRODUCTION

Cloud computing [1] is an exceptionally new model so there is no single definition has been acknowledged by the cloud clients. Various analysts gives number of meaning of distributed computing is by them planned. Be that as it may, we consider the definition gave by NIST (National Institute of guidelines and innovation) Information Technology Laboratory is as per the following:

Cloud computing is delivering administrations by decreasing knowledge ownership, improving mobility, dexterity in market, lowering foundation costs, and making assets accessible in real time.

By definition, cloud computing is not a single invention, but rather a combination of many developments that enables a new path for IT growth. In a case where there are few staff accessible at the server farm, if the solicitation sent is greater than the server farm's cap, the general presentation is corrupted. In such instances, a load balancer is used to boost the server.

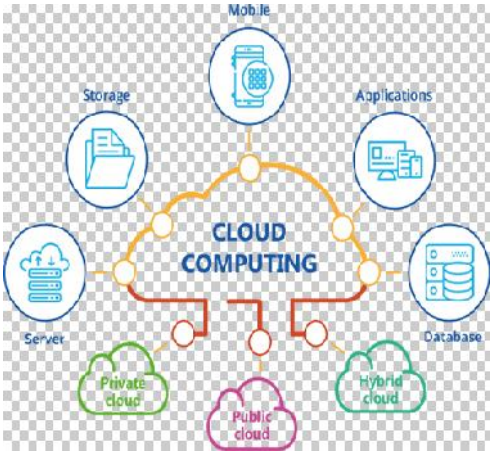

Figure 1: Cloud computing paradigm [3]

Load Balancing [2] [3] is a method to convey load among numerous substances, for example, CPUs, plate drives, worker or some other sort of gadget. The objective of Load Balancing is basically to acquire a lot more noteworthy use of assets. Load Balancing can be given either through equipment or programming. Load Balancing can be given through the specific gadgets, for example, a multilayer switch that can course the parcels to the objective or the bunch. Equipment based burden adjusting is unpredictable in arrangement and support, and not reasonable for facilitated climate.

Load Balancing can likewise be accomplished through the product either utilizing working framework or as an extra application. Programming based burden adjusting is easy to convey and have the exhibition like that of equipment based burden adjusting. Some product based burden offsetting incorporates those packs with Microsoft purplish blue or Linux and extra, for example, PM intermediary. Load balancer deals with the traffic stream between different workers. Load Balancer is set between the worker and the customer and appropriates the heap among the accessible workers relying on the calculation of the Load balancer. Load balancer isn't just improves the reaction season of cloud applications yet in addition guarantees the ideal use of the assets.

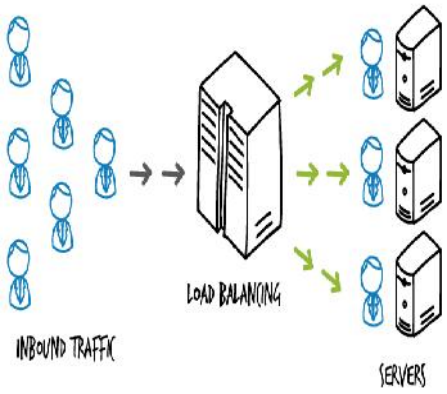

Figure 2: Load Balancing in Cloud
[source: https://www.ctl.io]

II. LITERATURE SURVEY

Specialists in (4, 5, 6) have talked about Round Robin (RR) booking calculation for load adjusting in a cloud situation. The premise of this calculation is the guideline of time booking. The Scheduler keeps up a rundown of accessible virtual machines in a table known as VM allotment table. It doles out the undertakings got through the datacenter controller to a rundown of virtual machines on pivot premise. Scheduler introduces the current_vm variable with the id of the main virtual machine. It maps the got errand with that VM whose id is put away in current_vm variable. On the off chance that estimation of current_vm is equivalent to id of last VM, at that point it initially instates current_vm with the id of first and do the mapping else it straightforwardly maps got task with that VM whose id is put away in current_vm variable

Specialists in (7, 8, 9) have talked about Throttled load adjusting procedure for cloud conditions. Throttled load balancer utilizes a solitary activity scheduler, which makes it incorporated in nature. The activity scheduler keeps up a table named VM portion table, which stores the id and status of all the virtual machines. A virtual machine can have just two states: involved or inert, meant by 1 or 0 separately in the cluster. At first, all virtual machines are inactive. On accepting an undertaking, work scheduler search the virtual machine which isn't occupied. On the off chance that it finds an inactive virtual machine, at that point it doles out the assignment to that virtual machine. On the off chance that no virtual machines are accessible to acknowledge the activity, at that point the errand needs to hold up in work scheduler's line. No lines are kept up at the virtual machine level. A virtual machine can oblige just one errand and another assignment can be apportioned just when the present undertaking has wrapped up

In (10, 11) specialists have talked about Equally Spread Current Execution (ESCE) load adjusting approach for cloud situations. This calculation utilizes the spread range approach. It works so that the quantities of dynamic assignments on each virtual machine are same whenever moment. The scheduler keeps up VM portion table which stores VM id and dynamic assignment depend on that VM. With the task of new undertakings or on task consummation, dynamic assignment include comparing to that VM in VM designation table will be refreshed. At the outset dynamic errand check of each VM is zero. On appearance of errand, ESCE scheduler finds that VM whose dynamic undertaking tallies is most reduced. On the off chance that more than one VM has most reduced dynamic tallies, at that point VM which has been recognized first is chosen for task. Undertaking lines are kept up comparing to each VM.

In the all-encompassing adaptation of ESCE, ESCE Scheduler occasionally breaks down the heap of virtual machines and reshuffles the heap to guarantee uniformity of burden by moving of burden from over-burden virtual machine to under-stacked virtual machine. Continued filtering of the line not just outcomes in the extra computational overhead, butalso brings about effective and even use of the heap. Another overhead connected with this all-encompassing form is determination of assignment to be relocated.

Scientists in (12, 13) have examined Minimum Completion Time (MCT) approach for load adjusting. Undertakings are alloted to assets in first start things out serve way. The virtual machine which sets aside less finish effort for a given errand is planned first. Finish time is assessed based on VM power and number of undertakings in VM line. Before all else, when no undertaking is assigned to VM then VM control is equivalent to its fulfillment time. For task of errand to a virtual machine, MCT Scheduler gets to the VM assignment table. VM distribution table stores the virtual machine id, virtual machine control, number of assignments in line and finish time of that virtual machine. This methodology is dynamic in nature as it thinks about the present heap of virtual machines.

Scientists in (14, 15) have talked about Minimum Execution Time (MET) approach for load adjusting. In this methodology, undertakings are doled out to assets in first start things out serve way. The virtual machine which takes less Execution Time (ET) for a given undertaking is booked first. Execution time is assessed based on preparing limit of virtual machines. MET Scheduler gets to the VM designation table for mapping of undertaking with VM. VM assignment table stores the virtual machine id and virtual machine handling limit. A virtual machine with additionally handling force can

execute the undertaking quick. Along these lines, this incorporated burden adjusting approach is static in nature which neither considers the present burden nor considers the errand size.

In (16, 17), specialists have talked about min-min approach of burden adjusting. This calculation doesn't pursue initially start things out serve arrangement rather it contains two criteria for task VM mapping:

- Minimum execution time
- Minimum finishing time

Least execution time assignments are favored over the most extreme execution time undertakings. It is settled based on task size. Errands are put away in the cushion. At the point when the cushion fills totally, at that point assignments are orchestrated in expanding request of the estimate and bunch is prepared. The idea picks the errand which holds least execution time and appoints it to the virtual machine which gives least consummation time. Least fulfillment time is assessed based on VM control and no. of assignments in the line of VM. It includes two least choice criteria, so it is called min-min approach.

Analysts in (18, 19) have examined Max-Min approach of burden adjusting. This calculation doesn't pursue initially start things out serve succession. It contains two criteria for task VM mapping:

- Maximum execution time
- Minimum culmination time.

Most extreme execution time undertakings are favored before the base execution time assignments. Assignments are put away in an undertaking allotment table till table fills totally. After this errand in the assignment distribution table are arranged in the diminishing request of their size. At that point the scheduler picks the undertaking which holds the greatest execution time. After this VM having least culmination time is chosen for task. Finish time is evaluated based on VM limit and no. of undertakings in the line of VM. It includes one most extreme and one least choice criteria, so it is called max min approach.

Analysts in (18, 19) have examined Max-Min approach of burden adjusting. This calculation doesn't pursue previously start things out serve succession. It contains two criteria for task VM mapping:

- Maximum execution time
- Minimum finish time.

Greatest execution time undertakings are favored before the base execution time errands. Errands are put away in an assignment allotment table till table fills totally. After this errand in the assignment portion table are arranged in the diminishing request of their size. At that point the scheduler picks the errand which holds the greatest execution time. After this VM having least culmination time is chosen for task. Finishing time is assessed based on VM limit and no. of undertakings in the line of VM. It includes one most extreme and one least choice criteria, so it is called max min approach.
Specialists in (20, 21) have talked about the join most brief line booking approach for load adjusting in an appropriated situation. This methodology utilizes just single scheduler, which keeps up the VM distribution table. VM distribution table stores VM id and the total load of dynamic undertakings doled out to that VM. At whatever point JSQ scheduler gets an undertaking, it advances the errand towards that virtual machine whose line length is little. The aggregate load of every id is utilized to demonstrate the line length. No lines are kept up at scheduler level.

Scientists in (22, 23) have talked about Join Idle Queue (JIQ) booking approach for load adjusting. JIQ was acknowledged utilizing two level booking. To understand the idea of two degrees of planning, creators has utilized the disseminated scheduler. Various schedulers are utilized. Quantities of schedulers are less in contrast with the quantity of virtual machines. Each scheduler will keep up a line of inert virtual machines. From the outset level, inactive VM is distinguished to be mapped with the assignment while at second level inert VM partners itself with any of the arbitrarily chosen scheduler.

On accepting an undertaking, scheduler initially counsels its inert line. On the off chance that it finds any virtual machine, which is inactive, at that point it promptly allocates the undertaking to that virtual machine and expels that virtual machine from its inert line. On the off chance that it doesn't locate any inactive virtual machine, at that point it arbitrarily maps the assignment with any VM.

Virtual machine, after occupation fulfillment, update about its status to any of the haphazardly picked inert lines related with a scheduler. This methodology isolates the errand of disclosure of inactive servers from the undertaking of employment task to a virtual machine. Because of the utilization of different schedulers, this methodology is appropriated in nature. Disappointment of one scheduler doesn't cause the disappointment of the whole framework.

III. CONCLUSION & FUTURE WORK

Load balancing is a technique to distribute load among multiple entities such as CPUs, disk drives, server or any other type of device. The goal of load balancing is primarily to obtain much greater utilization of resources. In this paper we have proposed a survey of load balancing methods. In cloud computing load balancing is one of the main issue. When client is requesting for service it should be available to the client. When any node is overloaded with job at that time load balancer has to set that load on another free node. Therefore load balancing is necessary in cloud computing. So in this paper we have discussed all the existing techniques for Load balancing.

The proposed strategy is actualized effectively and their presentation in various boundaries are assessed, as indicated by results the exhibition of the proposed procedure is adoptable and proficient in this manner the accompanying anticipated augmentations are conceivable with the proposed technique expansion. The computational intricacy as far as time multifaceted nature is required to upgrade in light of the fact that the time unpredictability of the framework is increments with the measure of information

REFERENCES

[1] Kansal NJ, Chana I. Cloud load balancing techniques: A step towards green computing. IJCSI International Journal of Computer Science Issues. 2012 Jan; 9(1):238-46.

[2] Katyal M, Mishra A. A comparative study of load balancing algorithms in cloud computing environment. arXiv preprint arXiv:1403.6918. 2014 Mar 27.

[3] Zhang Y, Franke H, Moreira JE, Sivasubramaniam A. A comparative analysis of space-and time-sharing techniques for parallel job scheduling in large scale parallel systems. IEEE Transactions on Parallel and Distributed Systems. 2002.

[4] Jayarani R, Sadhasivam S, Nagaveni N. Design and implementation of an efficient two-level scheduler for cloud computing environment. In Advances in Recent Technologies in Communication and Computing, 2009. ARTCom'09. International Conference on 2009 Oct 27 (pp. 884-886). IEEE.

[5] Wang SC, Yan KQ, Liao WP, Wang SS. Towards a load balancing in a threelevel cloud computing network. In Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on 2010 Jul 9 (Vol. 1, pp. 108-113). IEEE.

[6] Xu G, Pang J, Fu X. A load balancing model based on cloud partitioning for the public cloud. Tsinghua Science and Technology. 2013 Feb; 18(1):34-9.

[7] Mahajan K, Makroo A, Dahiya D. Round robin with server affinity: a VM load balancing algorithm for cloud based infrastructure. Journal of information processing systems. 2013; 9(3):379-94.

[8] Chaudhary D, Kumar B. Analytical study of load scheduling algorithms in cloud computing. In Parallel, Distributed and Grid Computing (PDGC), 2014 International Conference on 2014 Dec 11 (pp. 7-12). IEEE.

[9] Tyagi V, Kumar T. ORT Broker Policy: Reduce Cost and Response Time Using Throttled Load Balancing Algorithm. Procedia Computer Science. 2015 Dec 31; 48:217-21.

[10] Mohapatra S, Rekha KS, Mohanty S. A comparison of four popular heuristics for load balancing of virtual machines in cloud computing. International Journal of Computer Applications. 2013 Jan 1; 68(6).

[11] Zaouch A, Benabbou F. Load Balancing for Improved Quality of Service in the Cloud. International Journal of Advanced Computer Science and Applications IJACSA. 2015; 6(7):184-9.

[12] Xu G, Pang J, Fu X. A load balancing model based on cloud partitioning for the public cloud. Tsinghua Science and Technology. 2013 Feb; 18(1):34-9.

[13] Bagwaiya V, Raghuwanshi SK. Hybrid approach using throttled and ESCE load balancing algorithms in cloud computing. In Green Computing Communication and Electrical Engineering (ICGCCEE), 2014 International Conference on 2014 Mar 6 (pp. 1-6). IEEE.

[14] Ritchie G, Levine J. A fast, effective local search for scheduling independent jobs in heterogeneous computing environments.

[15] Hung CL, Wang HH, Hu YC. Efficient load balancing algorithm for cloud computing network. In International Conference on Information Science and Technology (IST 2012), April 2012 Apr 28 (pp. 28-30).

[16] Armstrong R, Hensgen D, Kidd T. The relative performance of various mapping algorithms is independent of sizable variances in run-time predictions. In Heterogeneous Computing Workshop, 1998 (HCW 98) Proceedings. 1998 Seventh 1998 Mar 30 (pp. 79-87). IEEE.

[17] Braun TD, Siegel HJ, Beck N, Bölöni LL, Maheswaran M, Reuther AI, Robertson JP, Theys MD, Yao B, Hensgen D, Freund RF. A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems. Journal of Parallel and Distributed computing. 2001 Jun 30; 61(6):810-37.

[18] Chen H, Wang F, Helian N, Akanmu G. User-priority guided Min-Min scheduling algorithm for load balancing in cloud computing. In Parallel Computing Technologies

(PARCOMPTECH), 2013 National Conference on 2013 Feb 21 (pp. 1-8). IEEE.

[19] Priya SM, Subramani B. A new approach for load balancing in cloud computing. International Journal of Engineering and Computer Science ISSN. 2013 May: 2319-7242.

[20] Miriam DD, Easwarakumar KS. A double min min algorithm for task metascheduler on hyper cubic p2p grid systems. International Journal of Computer Science Issues. 2010 Jul; 7(4):8-18.

[21] Maheswaran M, Ali S, Siegel HJ, Hensgen D, Freund RF. Dynamic mapping of a class of independent tasks onto heterogeneous computing systems. Journal of parallel and distributed computing. 1999 Nov 30; 59(2):107-31.

[22] Gupta V, Balter MH, Sigman K, Whitt W. Analysis of join-the-shortest-queue routing for web server farms. Performance Evaluation. 2007 Oct 31; 64(9):106281.

[23] Lin HC, Raghavendra CS. An analysis of the join the shortest queue (JSQ) policy. In Distributed Computing Systems, 1992, Proceedings of the 12th International Conference on 1992 Jun 9 (pp. 362-366). IEEE.

[24] Regina Miseviciene, Rimantas Plestys, Rokas Zakarevicius, Nerijus Pazereckas "Virtual Desktop Infrastructure Technology Based Study & Research" International Journal Of Education And Research Vol. 1 No. 4 April 2013.

[25] Ritu Kapur "A Cost Effective approach for Resource Scheduling in Cloud Computing" IEEE International Conference on Computer, Communication and Control (IC4-2015).

[26] Singh, SimarPreet, Anju Sharma, and Rajesh Kumar. "Analysis of Load Balancing Algorithms using Cloud Analyst." International Journal of Grid and Distributed Computing 9, no. 9 (2016): 11-24.

[27] Somula, R., &Sasikala, R. (2018). Round robin with load degree: An algorithm for optimal cloudlet discovery in mobile cloud computing. Scalable Computing: Practice and Experience, 19(1), 39-52. 9. Somula, R. S., &Sasikala, R. (2018).

[28] Somula, R., &Sasikala, R. (2019). A Honey Bee Inspired Cloudlet Selection for Resource Allocation. In Smart Intelligent Computing and Applications (pp. 335-343). Springer, Singapore.