International Journal of Engineering Science Invention (IJESI)
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org ||Volume 8 Issue 01 Series. III || Jan 2019 || PP 35-39

Selection of a Scheduler (Dispatcher) within a Datacenter using Enhanced Equally Spread Current Execution (EESCE)

G.Thejesvi[1], Dr.T.Anuradha[2]
[1]*(Research scholar, Dept of Computer Science, Dravidian University, Kuppam-517425, A.P)*
[2]*(Professor, Dept of Computer Science, Dravidian University, Kuppam-517425, A.P)*
Corresponding Author: G.Thejesvi

ABSTRACT*: Cloud Computing is one of the emerging technologies for the modern world. Cloud has lot of job opportunities as well as research purpose. The main challenges in the cloud computing are load balancing, security issues, and availability of resources. In our paper, we focused on load balancing in cloud by adding enhancements in the equally spread current execution algorithm. The main objectives of this algorithm are to distribute the work load equally among the all virtual machines. The problems in equally spread current execution algorithm were overcome by doing some enhancements in the existing algorithm. This proposed Enhanced Equally Spread Current Execution (EESCE) is used for selecting the best dispatcher (Scheduler) among the several dispatchers present in a single datacenter is discussed clearly in this paper.*

KEYWORDS – *Load balancing, equally spread current execution, scheduler (dispatcher), virtual machines*

Date of Submission: 20-01-2019 Date of acceptance: 06-02-2019

I. INTRODUCTION

IT industry is growing each day and so is the need for computing and storage resources. Large quantities of data are generated and exchanged over the network which further necessitates the need of more and more computing resources. Organizations, to better capitalize their investment, are opening their infrastructure to new found virtualization technologies like Cloud computing. Cloud has helped enterprises leverage the benefits of computing resources which are shared over a virtualized environment. A lot of enterprises are already using cloud-based services in one or the other form. This brings us to the concept of load balancing in cloud. A website or a web-application can be accessed by a plenty of users at any point of time. It becomes difficult for a web application to manage all these user requests at one time. It may even result in system breakdowns. For a website owner, whose entire work is dependent on his portal, the sinking feeling of website being down or not accessible also brings lost potential customers.

Cloud Load balancing is the process of distributing workloads and computing resources across one or more servers. This kind of distribution ensures maximum throughput in minimum response time. The workload is segregated among two or more servers, hard drives, network interfaces or other computing resources, enabling better resource utilization and system response time. Thus, for a high traffic website, effective use of cloud load balancing can ensure business continuity. The common objectives of using load balancers are:[1]

- To maintain system firmness.
- To improve system performance.
- To protect against system failures.

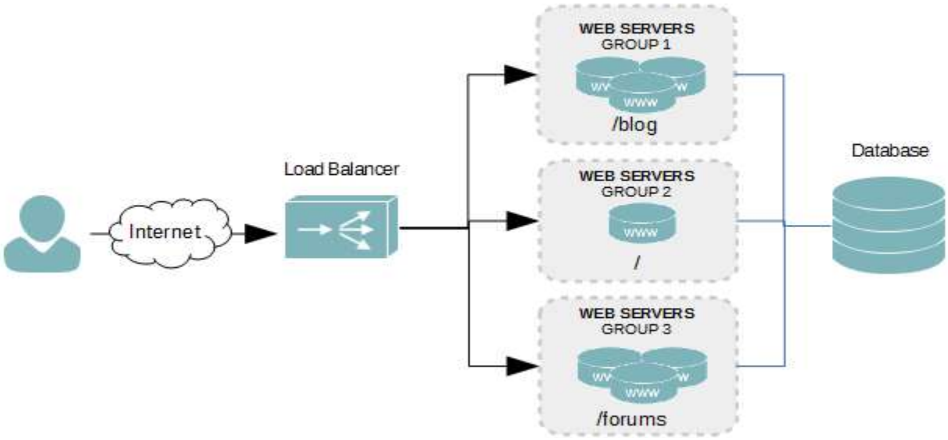
Cloud providers like **Amazon Web Services (AWS)**, **Microsoft Azure** and **Google** offer cloud load balancing to facilitate easy distribution of workloads. For ex: **AWS offers Elastic Load balancing (ELB) technology** to distribute traffic among EC2 instances. Most of the AWS powered applications have ELBs installed as key architectural component.
Similarly, **Azure's Traffic Manager** allocates its cloud servers' traffic across multiple datacenters.

II. How does load balancing work? [1]

Here, load refers to not only the website traffic but also includes CPU load, network load and memory capacity of each server. A load balancing technique makes sure that each system in the network has same amount of work at any instant of time. This means neither any of them is excessively over-loaded, nor under-utilized.

The load balancer distributes data depending upon how busy each server or node is. In the absence of a load balancer, the client must wait while his process gets processed, which might be too tiring and demotivating for him. Various information like jobs waiting in queue, CPU processing rate, job arrival rate etc. are exchanged between the processors during the load balancing process. Failure in the right application of load balancers can lead to serious consequences, data getting lost being one of them.

Different companies may use different load balancers and multiple load balancing algorithms like static and dynamic load balancing. One of the most commonly used methods is **Round-robin load balancing.** It forwards client request to each connected server in turn. On reaching the end, the load balancer loops back and repeats the list again. The major benefit is its ease of implementation. The load balancers check the system heartbeats during set time intervals to verify whether each node is performing well or not.



III. Advantages of Cloud Load Balancing [1]:

a) High Performing applications:

Cloud load balancing techniques, unlike their traditional on-premise counterparts, are less expensive and simple to implement. Enterprises can make their client applications work faster and deliver better performances, that too at potentially lower costs.

b) Increased scalability

Cloud balancing takes help of cloud's scalability and agility to maintain website traffic. By using efficient load balancers, you can easily match up the increased user traffic and distribute it among various servers or network devices. It is especially important for ecommerce websites, who deals with thousands of website visitors every second. During sale or other promotional offers they need such effective load balancers to distribute workloads.

c) Ability to handle sudden traffic spikes

A normally running University site can completely go down during any result declaration. This is because too many requests can arrive at the same time. If they are using cloud load balancers, they do not need to worry about such traffic surges. No matter how large the request is, it can be wisely distributed among different servers for generating maximum results in less response time.

d) Business continuity with complete flexibility

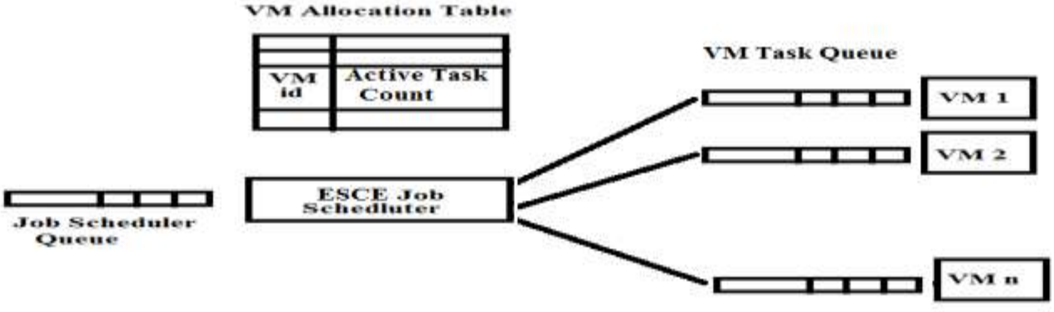
The basic objective of using a load balancer is to save or protect a website from sudden outages. When the workload is distributed among various servers or network units, even if one node fails the burden can be shifted to another active node.

Thus, with increased redundancy, scalability and other features load balancing easily handles website or application traffic.

IV. Equally Spread Current Execution:

In Equally Spread Current Execution (ESCE) load balancing approach for cloud environments. This algorithm uses the spread spectrum approach. It works in such a way that the numbers of active tasks on each virtual machine are same at any time instant. The scheduler maintains VM allocation table which stores VM id and active task count on that VM. With the assignment of new tasks or on task completion, active task count corresponding to that VM in VM allocation table will be updated. In the beginning active task count of each VM is zero. On arrival of task, ESCE scheduler finds that VM whose active task counts is lowest. If more than one VM has lowest active counts, then VM which has been identified first is selected for task assignment. Task queues are maintained corresponding to each VM.

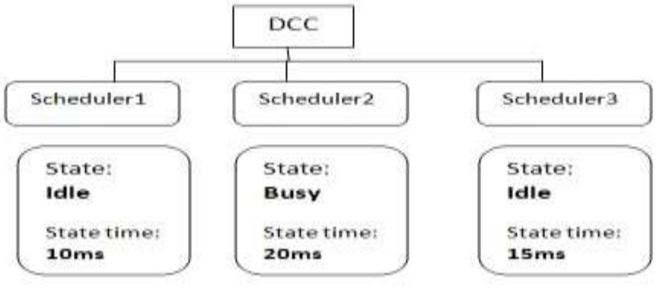
In the extended version of ESCE, ESCE Scheduler periodically analyzes the load of virtual machines and reshuffles the load to ensure equality of load by transferring of load from overloaded virtual machine to under-loaded virtual machine. Repeated scanning of the queue not only results in the additional computational overhead, but also results in efficient and even utilization of the load. Another overhead associated with this extended version is selection of task to be migrated. Also, while mapping task with VM, if task size and VM capacity are considered then this will result in another extended version of equally spread current execution load balancing technique. The extra overhead in this approach is in terms of calculation and storage of task size and VM capacity. Below figure shows the idea of equally spread current execution approach.



Methodology:

Selection of a Scheduler based on EESCE:

1. Every datacenter has datacenter controller (DCC) which maintains the information of all the available schedulers in that datacenter.
2. The Datacenter Controller (DCC) maintains scheduler information such as scheduler state (Idle/ Busy) and state time.



3. Scheduler state idle or busy can be determined by as follows:
3.1: Scheduler state=**"idle"**

When (Idle Queue Length > = Waiting Queue Length)

Where,
Idle queue length= no. of Virtual machines (vm) in the idle queue.
Waiting queue length = no. of jobs waiting in the waiting queue.

3.2: Scheduler state="**Busy**"

When (Idle Queue Length < Waiting Queue Length)

Where,
Idle queue length= no. of Virtual machines (vm) in the idle queue.
Waiting queue length = no. of jobs waiting in the waiting queue.

4. Datacenter Controller (DCC) checks all the available scheduler's states and its state time, and selects one scheduler as follows.

Scenario 1:

A. If two or more Schedulers' state is "IDLE" with different point of time for the states
- DCC will select a scheduler with longer idle time.

Scenario 2:

B. If more than one Schedulers state is "IDLE" with same state time.
- DCC will select one scheduler randomly.

Scenario 3:

C. If two or more Schedulers state is "Busy" and job count in waiting queue is different.
- DCC will select a Scheduler based on Equally Spread current Execution (ESCE).
- Equally spread current execution works on the logic of maintaining equal work load among all the schedulers.
- DCC maintains all Schedulers information like waiting Queue length and no. of active jobs allotted to this particular scheduler.
- Based on this information, Datacenter Controller will select a scheduler with least count of jobs in the waiting queue.

Scenario 4:

D. If two or more Schedulers state is "Busy" and job count in waiting queue is also same.
Data Center Controller will select one scheduler randomly among all the available schedulers.

Algorithm:

Algorithm EESCE ( )
{
Datacenter Controller maintains the active job count of all schedulers;
While (data center controller receives a new task)

```
{
        For (i=1 to n)
        If (Schedulers state were idle) then
        {
                If (state time of all idle schedulers were also same) then
                        {
                                DCC will selects one scheduler randomly;
                        }
                Else
                        {
                                DCC will selects a scheduler with longer idle time;
                        }
        }
If (Schedulers state were busy) then
        {
                If (waiting queue length of all busy scheduler were same) then
                {
                        DCC will select one scheduler randomly;
                }
                Else
                {
                Identify the scheduler whose waiting queue length is least;
                Assigns the new job to that scheduler;
                Update the waiting queue length by incrementing 1;
                }
           }
}
}
}
```

V. Conclusion:

The enhanced equally spread current execution algorithm is implemented in cloud analyst simulator .This load balancing algorithm compares with remaining load balancing algorithms like round robin, throttled algorithms and hence shown better performance in finding the best & optimized scheduler in a particular data center.

REFERENCES

[1]. https://www.znetlive.com/blog/what-is-load-balancing-in-cloud-computing-and-its-advantages/.
[2]. A.Khiyaita, M. Zbakh, H. El Bakkali and Dafir El Kettani, "Load Balancing Cloud Computing: State of Art" , 9778-1-4673-1053-6/12/$31.00, 2012 IEEE.
[3]. A.M. Alakeel, "A Guide to dynamic Load balancing in Distributed Computer Systems", International Journal of Computer Science and Network Security (IJCSNS), Vol. 10, No. 6, June 2010, pages 153-160.
[4]. Mladen A. Vouk, Cloud Computing Issues, Research and Implementations, Proceedings of the ITI 2008 30th Int. Conf. on Information Technology Interfaces, 2008, June 23-26.
[5]. Zhong Xu, Rong Huang,(2009)"Performance Study of Load Balanacing Algorithms in Distributed Web Server Systems", CS213 Parallel and Distributed Processing Project Report.
[6]. A. Bhadani , And S. Chaudhary , "Performance Evaluation Of Web Servers Using Central Load Balancing Policy Over Virtual Machines On Cloud", Proceedings Of The Third Annual ACM Bangalore Conference (COMPUTE), January 2010.
[7]. G.Thejesvi, Dr.T.Anuradha, "Distribution of Work Load at Main Controller Level using Enhanced Round Robin Scheduling Algorithm in a Public Cloud" Published in International Journal of Computer Science and Engineering (IJCSE) Volume-3, Issue-12, Dec 2015 .
[8]. G.Thejesvi, Dr.T.Anuradha "Proposed Model for Proper Balancing of Load in the Public Cloud" Published in International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE) Volume-6, Issue-8, August 2016 .
[9]. G.Thejesvi, Dr.T.Anuradha, "Distribution of Work Load at Balancer Level using Enhanced Round Robin Scheduling Algorithm in a Public Cloud" Published in International Journal of Advances in Electronics and Computer Science (IJAECS) Volume-3, Issue-2, Feb 2016 .

G.Thejesvi" Selection of a Scheduler (Dispatcher) within a Datacenter using Enhanced Equally Spread Current Execution (EESCE)" International Journal Of Engineering Science Invention (Ijesi), Vol. 08, No. 01, 2019, Pp 35-39