

# Framework to Solve Response Time in Heterogeneous Server using Load Balancer

Charu Nasre<sup>1</sup> Shubham Nandane<sup>2</sup> Ankit Ambekar<sup>3</sup> Piyush Tiwaskar<sup>4</sup> Prof.S.M.Malode<sup>5</sup>

<sup>1,2,3,4</sup>Department of Information & Technology

<sup>1,2,3,4,5</sup>Zagdu Singh Charitable Trust's Thakur Polytechnic, Kandivali (E), Mumbai, India

**Abstract**— The project titled “Framework to Solve Response Time Problem in Heterogeneous Server Using Load Balancing.” The aim of project is to distribute the requests coming from multiple user for a particular site, in order to reduce congestion. To reduce such type of traffic the Load Balancer is used. Cloud computing is the use of various services, such as software development platforms, servers, storage and software, over the internet, often referred to as the cloud. The project is based on AWS i.e Amazon Web Services platform. The main objective behind this project is to increase the performance significantly & to reduce the response time of user.

**Key words:** Cloud Computing, Elastic Load Balancer, Auto Scaling, Elastic Beanstalk

## I. INTRODUCTION

Cloud computing is the use of various services, such as software development platforms, servers, storage and software, over the internet, often referred to as the "cloud."

In general, there are three cloud computing characteristics that are common among all cloud-computing vendors:

The back-end of the application (especially hardware) is completely managed by a cloud vendor.

A user only pays for services used (memory, processing time and bandwidth, etc.). Services are scalable

Many cloud computing advancements are closely related to virtualization. The ability to pay on demand and scale quickly is largely a result of cloud computing vendors being able to pool resources that may be divided among multiple clients.

It is common to categorize cloud computing services as Infrastructure as a Service (IaaS), Platform as a Service (PaaS) or Software as a Service (SaaS).

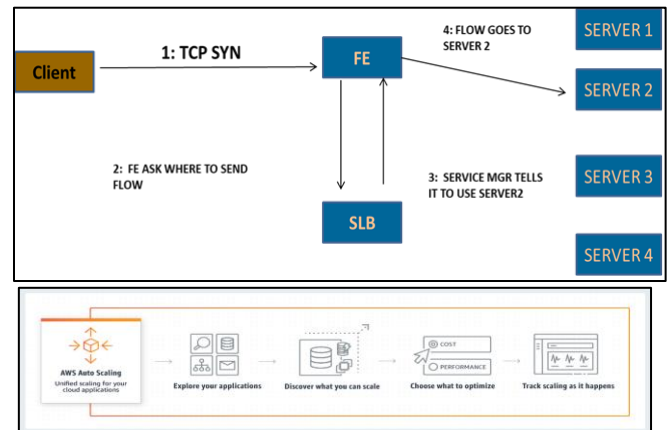
### A. Objective

- 1) To increase the performance significantly.
- 2) To have a backup plan in case the system fails even partially.
- 3) To keep the system stable.
- 4) To provide future enhancement in the system

### B. Problem Statement

- 1) Load on particular server increase and time requires to process Request more
- 2) Congestion will be more and most of the request will get canceled out

## II. BLOCK DIAGRAM



### A. Description:

Block diagram of a distributed architecture to perform a distributed system in load balancing. Whose components are located on different networked computers? Which communicate and coordinate their actions by passing messages to one another.

FE: Forwarding engine which are responsible for forwarding packet they ask the SLB (server load balancing) device where to send the flow

## III. LITERATURE REVIEW

### A. Load Balancing in Structured P2P System:

This cloud results is an  $O(\log N)$  imbalance. This paper address the problem of load balancing in such P2P systems .the simulation result show that even the simplest scheme is able to balance the load within 80% of the optimal value ,while the most complex scheme is able to balance the 95% of the optimal value.

### B. Comparative Study of Load Balancing Algorithm in Cloud Computing Environment:

Lo scheduling criteria. This ad balancing may even support prioritizing users by applying appropriate paper presents various load balancing schemes in different cloud environment based on requirements specified in service level agreement (SLA).

### C. Load Balancing and its Algorithms in Cloud Computing:

This paper mainly focuses on the concept of load balancing, literature survey on load balancing techniques and different measurement parameter.

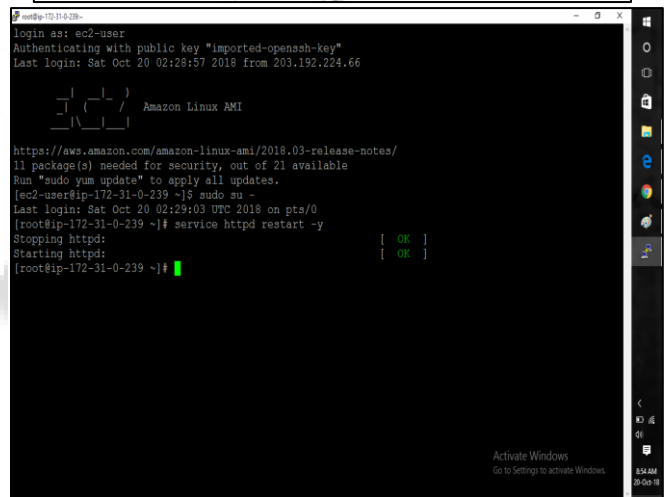
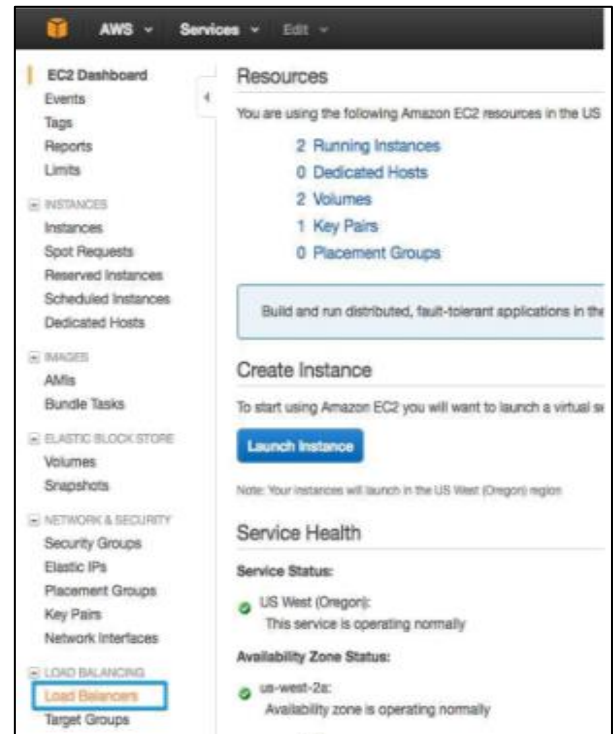
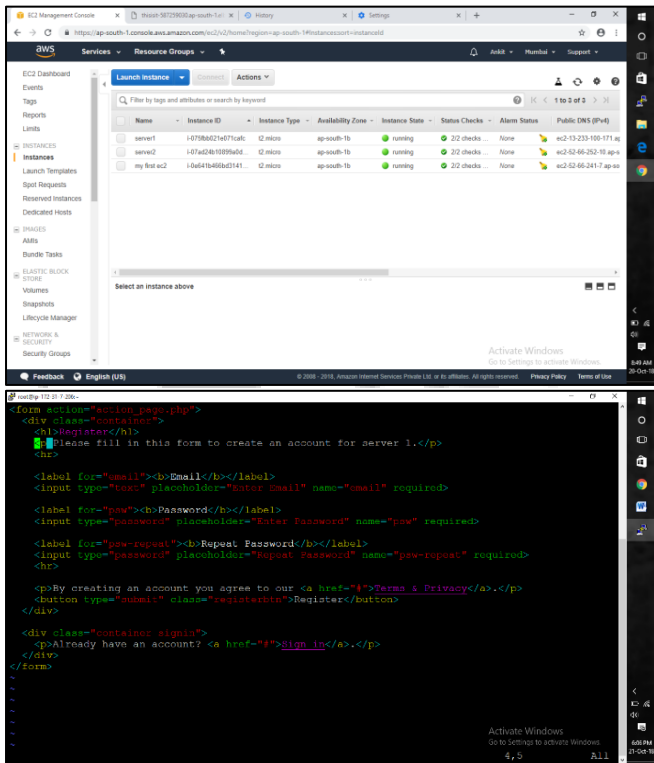
## IV. PROPOSED SYSTEM

A load balancer accepts incoming traffic from clients and routes requests to its registered targets (such as EC2 instances) in one or more Availability Zones. The load

balancer also monitors the health of its registered targets and ensures that it routes traffic only to healthy targets. When the load balancer detects an unhealthy target, it stops routing traffic to that target, and then resumes routing traffic to that target when it detects that the target is healthy again. You configure your load balancer to accept incoming traffic by specifying one or more *listeners*. A listener is a process that checks for connection requests.

It is configured with a protocol and port number for connections from clients to the load balancer and a protocol and port number for connections from the load balancer to the targets. Elastic Load Balancing supports three types of load balancers: Network Load Balancers, and Classic Load Balancers. There is a key difference between the way you configure these load balancers. With Application Load Balancers and you register targets in target groups, and route traffic to the target groups. With Classic Load Balancers, you register instances with the load balancer.

A. Screenshot of Result:



V. TECHNOLOGIES USED COMPUTER VISION:

Elastic Load Balancing offers two types of load balancers both of which feature high availability, automatic scaling, and robust security. These include the Classic Load Balancer that routes traffic based on either application or network Level information, and the Application Load Balancer that routes traffic based on advanced application level information that includes the content of the request.

A. Database Technologies:

AWS Database using like ERP, CRM, and ecommerce to log transactions and store structured data. Internet scale applications like hospitality, dating, and ride sharing to serve content and store structured and unstructured data.

Analytic applications for operational reporting and querying Terabyte to Exabyte scale data.

Real-time application use cases that require sub-millisecond latency like gaming leader boards, chat/messaging, streaming, and IoT.

Applications with use cases that require navigation of highly connected data like social news feeds, recommendations, and fraud detection.

1) RDS:

Amazon Relational Database Service (Amazon RDS) is a web service that makes it easier to set up, operate, and scale a relational database in the cloud. It provides cost-efficient, resizable capacity for an industry-standard relational database and manages common database administration tasks.

## VI. CONCLUSION

Hence in AWS ELB the load balancer distributes traffic evenly across the availability zones that you enable for your load balancer. AWS elastic load balance is one of the effective ways to manage and deploy the application as it is fast, reliable and efficient

## REFERENCES

- [1] M. Katyal and A. Mishra, "A comparative study of load balancing algorithm in cloud computing environment" in international journal of distributed and cloud computing. December 2013
- [2] Ananth Rao, Karthik Lakshminarayanan, Sonesh Surana, Richard Karp, and Ion Stoica, "Load Balancing in Structured P2P Systems," in Proc. IPTPS, Feb. 2003.
- [3] David Karger and Matthias Ruhl, "New Algorithms for Load Balancing in Peer-to-Peer Systems," Tech. Rep. MIT-LCS-TR-911, MIT LCS, July 2003.
- [4] Madhurima Rana, saurabh Lgaiyan, Ustav Kar, "A Study of Load Balancing in Cloud Computing Environment Using Evolutionary and swarm Based Algorithms" in international conference on control, Instrumentation, communication and computational Technologies (ICCICT) 2014
- [5] Peenaz Pathak, Kamna Mahajan, "A Review on Load Balancing in Cloud Computing" in International Journal Of Engineering And Computer Science (ISSN) 6 June 2015
- [6] Forum F Kherani, Prof. Jignesh Vania "Load Balancing in Cloud Computing" in International Journal Of Engineering Development and Research (IJEDR) 2014
- [7] Rahul Godha, sneh Prateek "Load Balancing in a Network" in International Journal Of Scientific and Research Publication, 10 oct 2014
- [8] Vinay Darji, Jayna Shah, Rutvik Mehta "Survey on Various load balancing algorithms in cloud computing" in International Journal Of Scientific & Engineering Research, 5 May 2014
- [9] Amit Gajbhiye, Dr. Shailender Singh "Global Server Load Balancing with Networked Load Balancers For Geographically Distributed cloud Data-Centers" in International journal of Computer science and Network (IJCSN), 6 Dec 2017
- [10] Sajjan R.S, Biradar Reklha Yashwantrao "Load Balancing and its Algorithms in cloud computing : A Survey" in International journal of Computer Science and Engineering (IJCSE), 6 Jan 2017