



Google Cloud

# Comparative Study on Load Balancing

## Google Cloud Platform VS. Amazon Web Service

Tirth Shah Piyush Thummar Jasmine Kaur

40115911

40125029

40103309

Concordia University, Montreal, Quebec



### Abstract

- High volume of traffic and user requests are essential reasons behind load balancing.
- Load balancer enhance responsiveness and increase the availability of applications.
- Provides efficient transfer of data over network.
- Optimizes application delivery and usage.
- Study on load balancing mechanism on Google Cloud Platform (GCP) and Amazon Web Service (AWS).
- Performance analysis of load management scenarios on GCP and AWS.

### Introduction

- Load Balancing is an efficient distribution of workloads among multiple computing resources.
- Provides maximum throughput in minimum response time.
- Idea of a comparative study is to scrutinize, relate and compare load management and efficiency of servers
- Statistical observations like Response Time, Latency, Transaction Rate and Throughput.
- Cloud Load Balancing can put the resources behind a single anycast IP and scale resources up or down with intelligent autoscaling.
- Cloud Load balancing integrated with Cloud CDN for optimal application and content delivery.
- It provides cross-region load balancing, including automatic multi-region failover, which gently moves traffic in fractions if backend become unhealthy.
- Amazon Web Service handles the load by Elastic Load Balancing.
- For the load balance of HTTP/HTTPS application, AWS uses layer 7- specific features, such as X-Forwarded-For headers.
- AWS Web Application Firewall to protect web applications on Application Load Balancers.

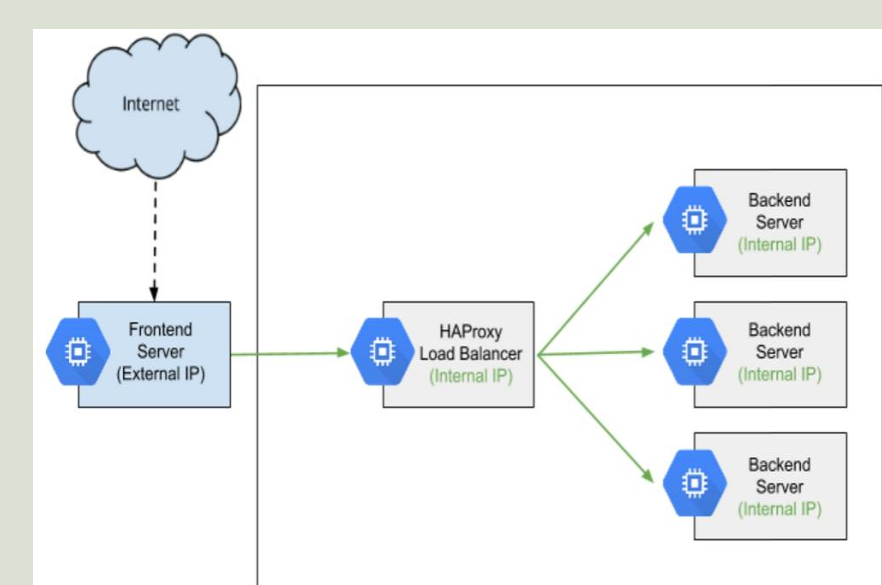


Figure 1. Google Cloud Platform.

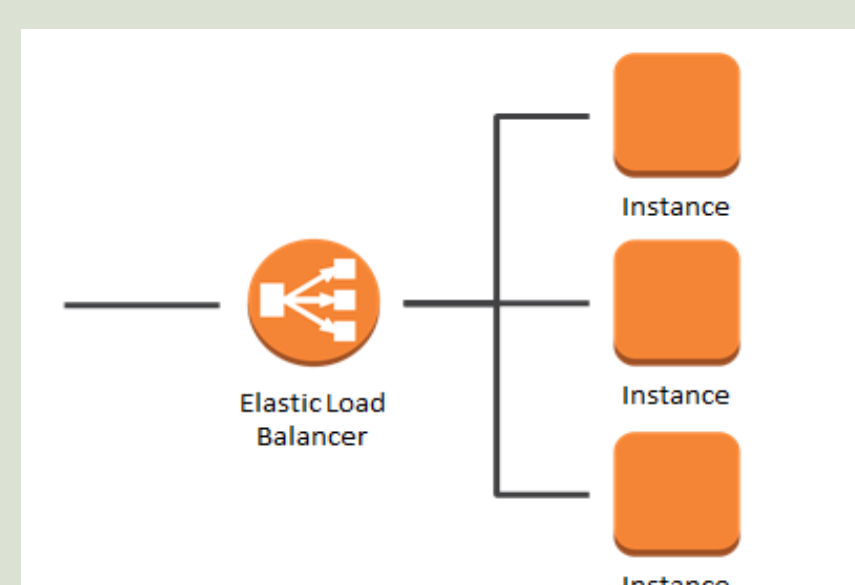


Figure 2. Amazon Web Service.

### Solution

- Performing the load test on the servers and evaluate them based on the parameters such as bandwidth, latency and throughput.
- In GCP, setting up the default compute engine zone, then create and configure Virtual Private Cloud (VPC).
- For the testing on the VM instances, set up the “load test” VM instance
- For the testing purpose, this VM instance uses a Python script to create a CPU-intensive task by calculating and displaying a picture of the Mandelbrot set on each request to the root (/) path.
- For AWS, create a Virtual Private Cloud (VPC) for communication among the cloud platform.
- In AWS, the security group and ACL (Network Access Control List) is being set up for the additional security against the incoming traffic.
- Setting up the target groups and performs load test.

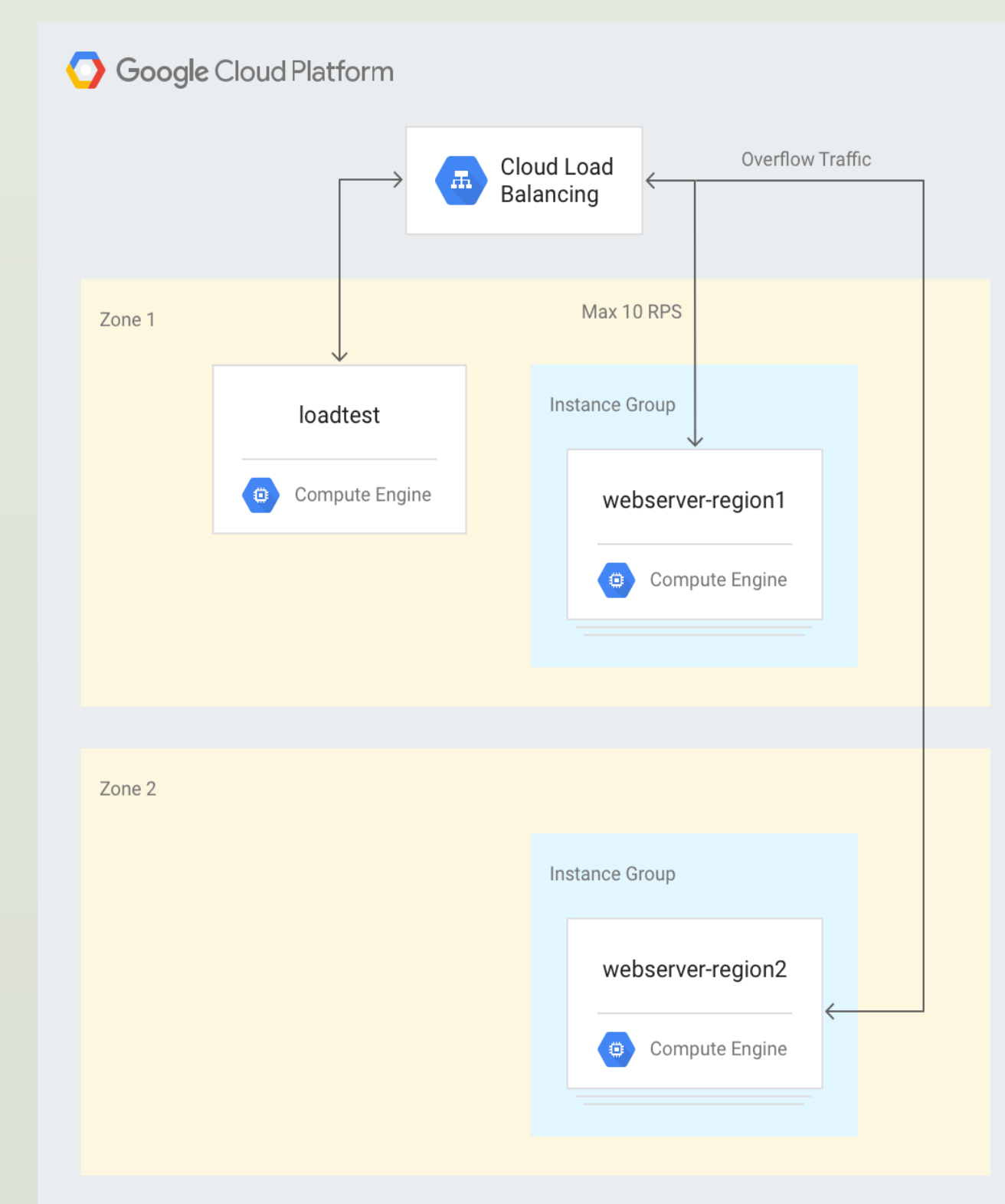


Figure 3. VM instances distributed in different zones handles by load balancer.

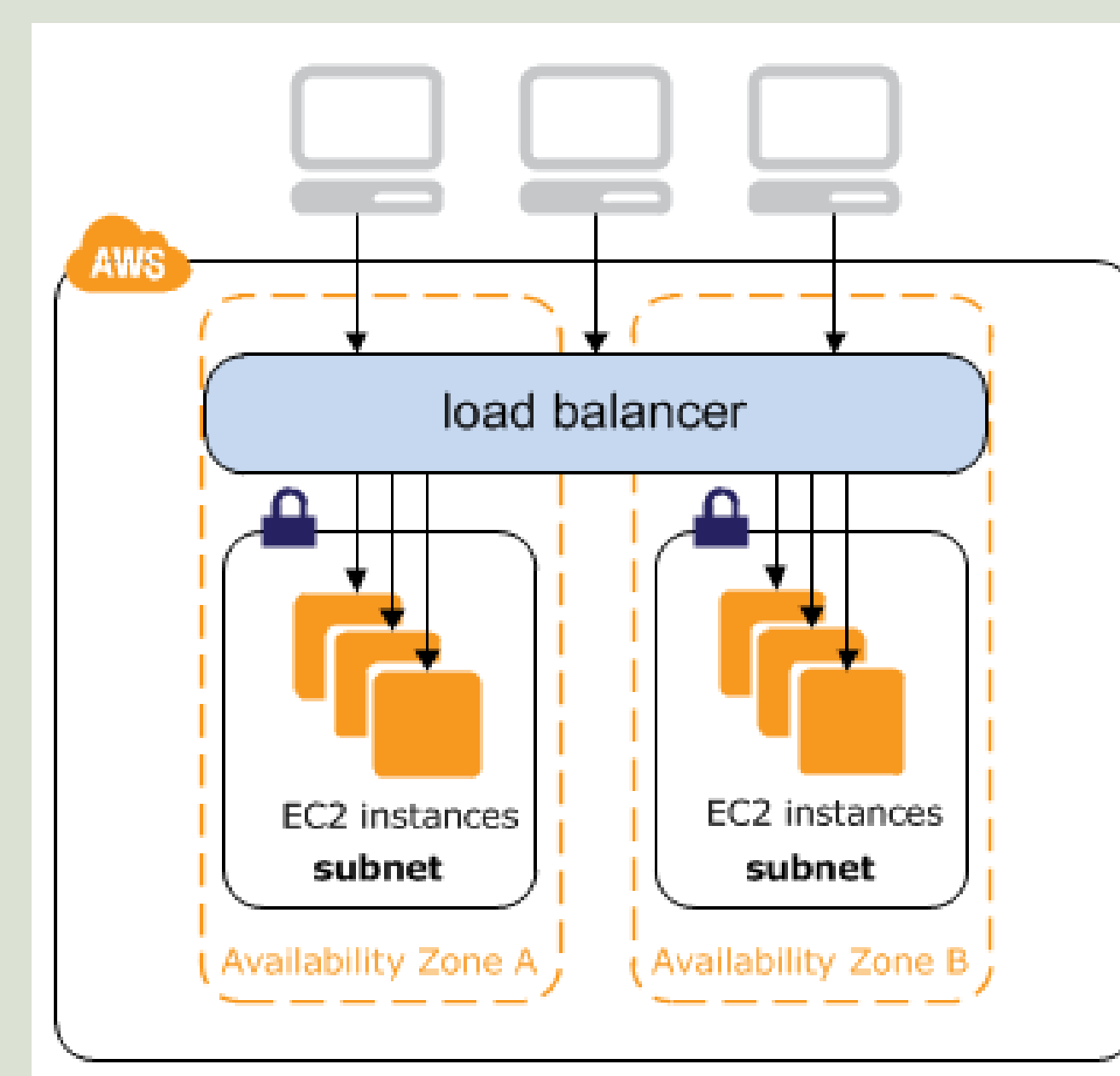


Figure 4. EC2 instances distributed in different zones handles by load balancer.

### Tools

## Siege Httpperf

- An open source regression test and benchmark utility
- Stress test a single URL with a user defined number of simulated users.
- Supports HTTP/1.0 and 1.1 protocols, the GET and POST directives, cookies, transaction logging, and basic authentication.

- Tool for measuring web server performance.
- Provides a flexible facility for generating various HTTP workloads and for measuring server performance.
- Robust and support for the HTTP/1.1 and SSL protocols.

### Result

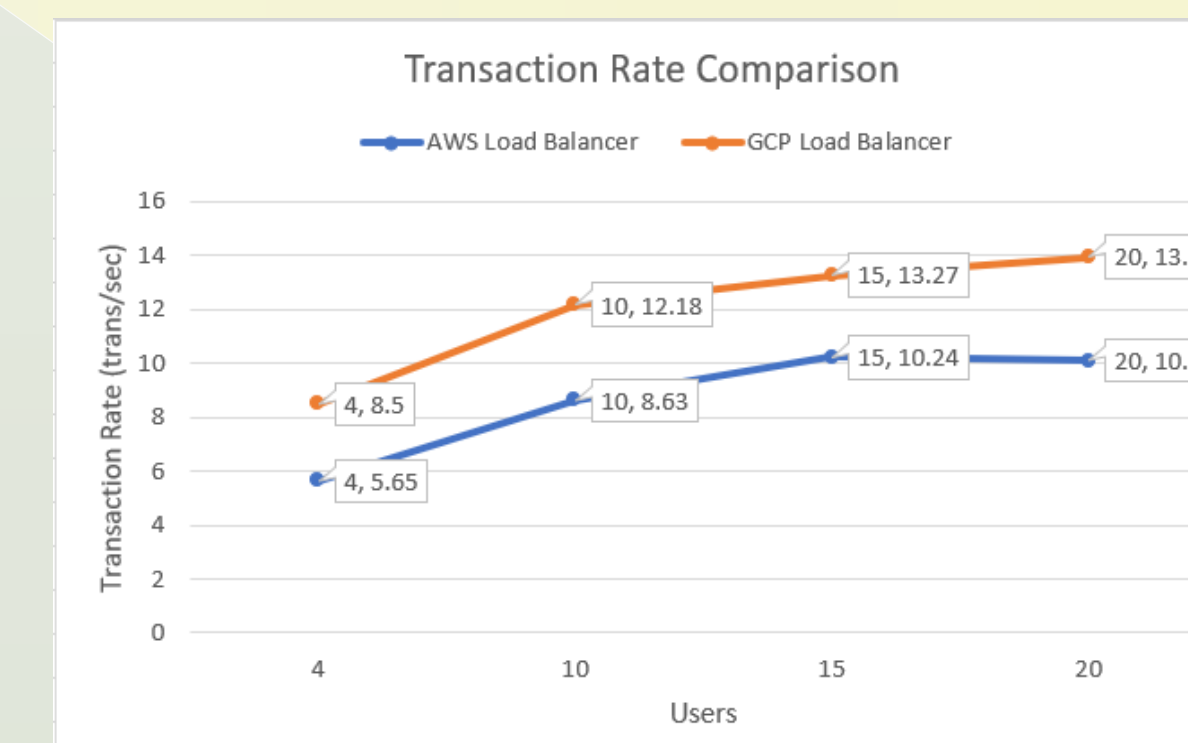


Chart 1. Transaction Rate.

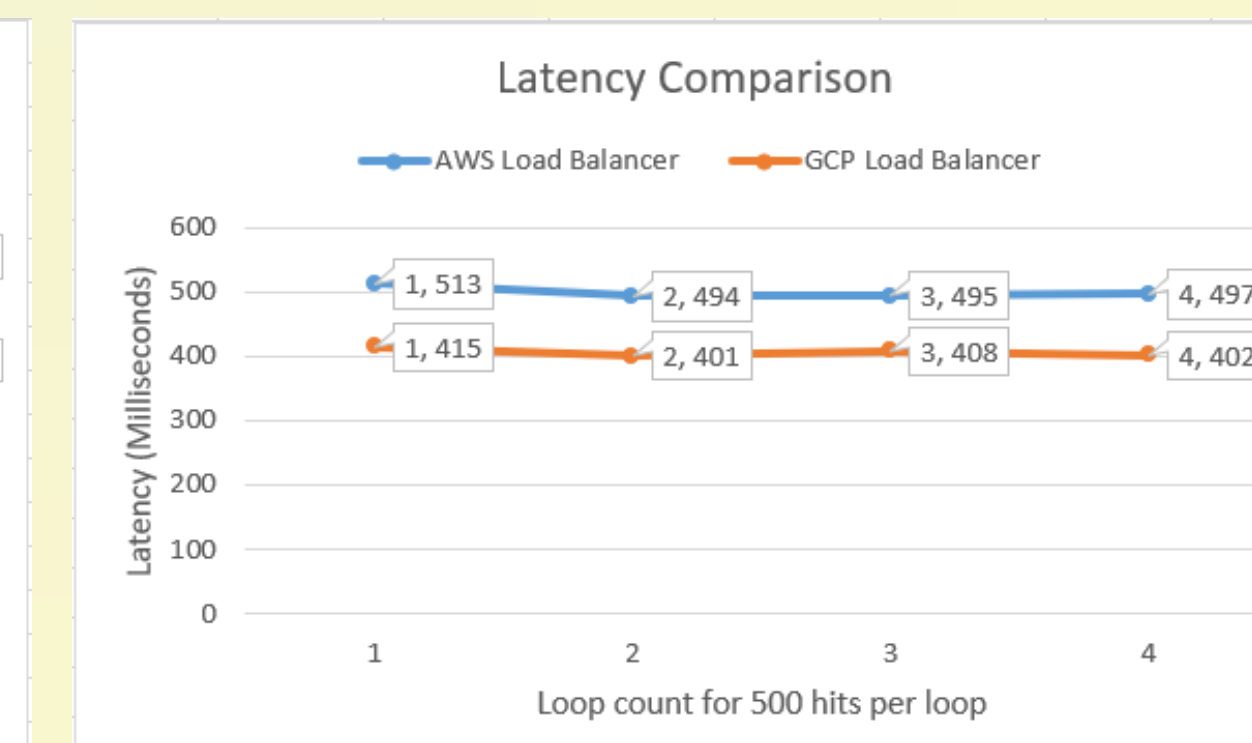


Chart 2. Latency.

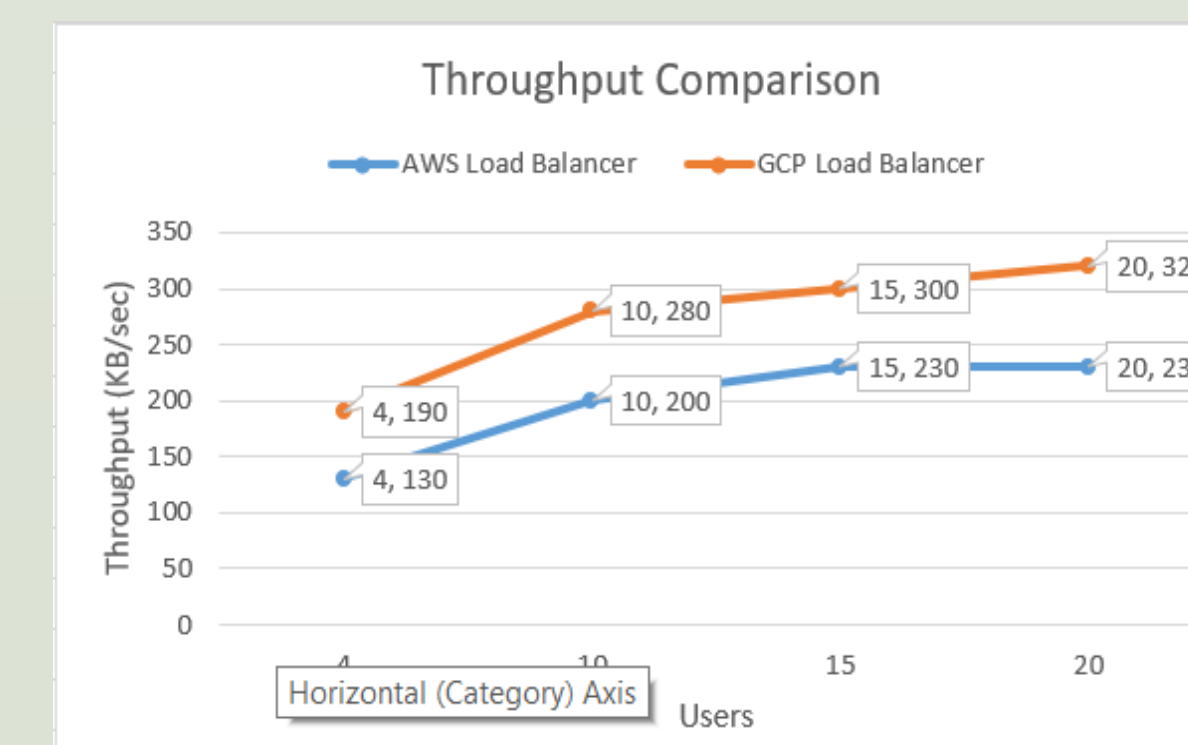


Chart 3. Throughput.

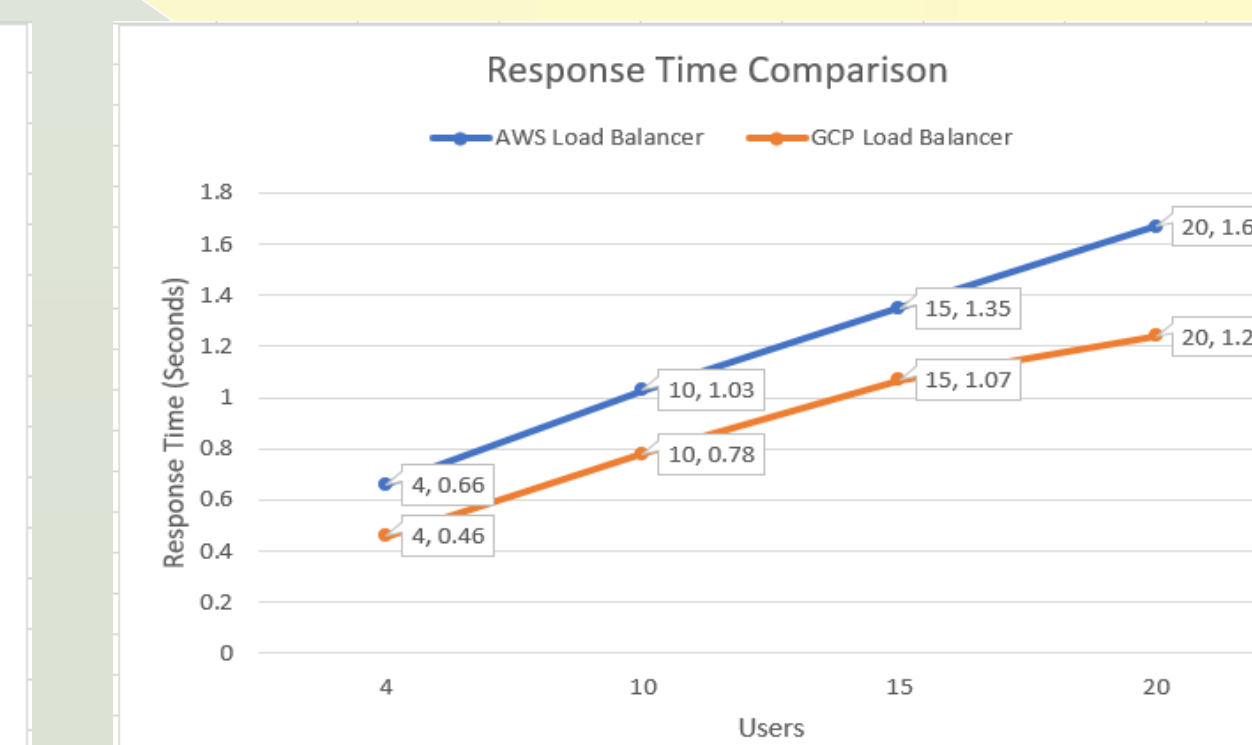


Chart 4. Response Time.

### Evaluation

Parameters	GCP	AWS
Latency	✓	✗
Throughput	✓	✗
Response Time	✓	✗
Transaction Rate	✓	✗

### Conclusion

- From analysis of GCP and AWS in terms of load balancing, GCP performed better than AWS.
- GCP have various configuration and protocols which supports to have efficient management of load than AWS.

### Future Work

- For enhancement of study, It should have dynamic load balancing. In which load balancer comes in action when load exceeds its limit.
- Consider network security measures of both platforms.
- Consider Microsoft Azure for further study.
- For GCP, latency will be optimize to increase the performance of the system.
- Extend this comparative study to every aspect of the Google cloud Platform and Amazon Web Service.

### References

1. Google Cloud Load Balancing. Retrieved from <https://cloud.google.com/load-balancing>
2. Elastic Load Balancing – Amazon Web Service. Retrieved from <https://aws.amazon.com/elasticloadbalancing/>
3. Rajguru, Abhijit A., and S. S. Apte. "A comparative performance analysis of load balancing algorithms in a distributed system using qualitative parameters." International Journal of Recent Technology and Engineering 1.3 (2012): 175-179.
4. James, Jasmin, and Bhupendra Verma. "Efficient VM load balancing algorithm for a cloud computing environment." International Journal on Computer Science and Engineering 4.9 (2012): 1658.
5. Jyoti Vashistha and Anant Kumar Jayswal "Comparative Study of Load Balancing Algorithms." International Organization of Scientific Research Vol. 3, Issue 3 (Mar. 2013).
6. Optimizing Application capacity with load balancing. Retrieved from <https://github.com/GoogleCloudPlatform/lb-app-capacity-tutorial-python#optimizing-application-capacity-with-global-load-balancing>
7. Optimizing application latency with load balancing. Retrieved from <https://cloud.google.com/load-balancing/docs/tutorials/optimize-app-latency>