NAME: Yağız SURNAME:Gürdamar   STUDENT NO: 22534

CS 436

HW #3

SERVER-SIDE LOAD BALANCING

In this homework you will implement the architecture given on Figure 1.

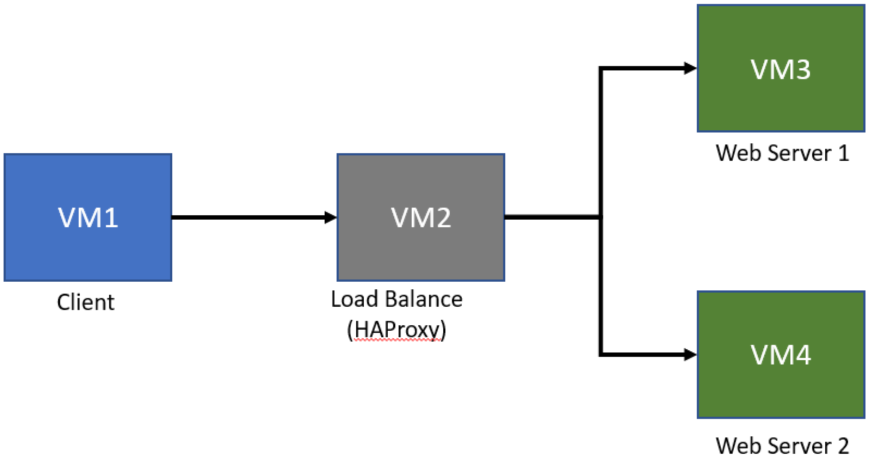

Figure 1 Architectural Overview

You need four VMs to achieve this.

STEP 1: Spin-off VMs

VM1,VM2,VM3,VM4: Ubuntu 22.04 Servers with 1GB RAM

STEP 2: Install necessary software

VM2: HAProxy https://www.linuxtechi.com/install-configure-haproxy-on-ubuntu/

VM1, VM3, VM4: Apache2

STEP 3:Test and see that all VMs can ping one another!

STEP 4: Basic test for load balancer

Run HAProxy load balancer and connect them to the servers. (You already installed it in STEP 3, so don't try to re-install when following the steps in the below link)

https://www.linuxtechi.com/install-configure-haproxy-on-ubuntu/

Make the test given in the link to see that requests are passed to both servers in a round robin manner.

STEP 5: Use ab tool to test the performance of your system

Use ab tool to individually test the servers and test the system with load balancing. Define what you consider as light, medium and heavy traffic in the table below:

	ab command with exact parameters
Light Traffic	ab -c 10 -n 20 http://172.26.37.129/ &
Medium Traffic	ab -c 100 -n 200 http://172.26.37.129/ &
Heavy Traffic	ab -c 1000 -n 2000000 http://172.26.37.129/ &

Now test your system with **heavy traffic** both with load balancer and without load balancer.

Enter the results below:

	Load Balancer (Client -> LB -> 2 web servers)	No Load balancer (Client -> single web server)
Latency (min)	1010	162
Latency (max)	79743	97095
Latency (mean)	1864	1127

Short summary of your findings:

Without Load Balancer:

- **Light Traffic**: Very low latency, with maximum latency only reaching 10 ms.
- **Medium Traffic**: Latency begins to increase, with maximum latency at 76 ms.
- **High Traffic**: Significant increase in latency, especially the maximum latency which spikes to 97095 ms.

With Load Balancer:

- **Light Traffic**: Slightly higher latency than without a load balancer, indicating a small overhead introduced by HAProxy.
- **Medium Traffic**: Increased latency, with the maximum reaching 1051 ms, showing that the load balancer is beginning to work under strain.
- **High Traffic**: Lower maximum latency compared to the no load balancer scenario, suggesting better distribution of requests among servers.

Findings Summary:

- The use of HAProxy introduces some overhead in lower traffic scenarios, as seen in the light traffic tests. This is expected since the load balancer adds an additional hop in the network path.
- In medium traffic scenarios, the overhead of the load balancer is more pronounced, with higher mean and maximum latencies compared to the direct server approach.
- However, under high traffic conditions, HAProxy significantly reduces the maximum latency, highlighting its effectiveness in distributing the load and preventing server overload. The mean and maximum latencies, while higher than in lower traffic levels, are much lower than the maximum latencies experienced without a load balancer.
- The extreme spike in maximum latency without a load balancer under heavy traffic suggests that the server becomes a bottleneck, potentially leading to timeouts and errors.
- Overall, HAProxy helps to maintain service availability and consistency under heavy load, even though it introduces a slight increase in latency under lighter loads. It appears to prevent the single server from being overwhelmed, which

could be crucial in a production environment where consistent response times are needed.