Modularization Workshop with Virtualization and Introduction to Docker and AWS

Johan Sebastian Arias

September 2020

1 Introduction

This workshop consists in creating an architecture that contains a load balancer, 3 nodes that handle REST requests and a node that contains a mongo database. Each node represents a container in Docker. The idea is to work with the Spark java micro-framework. Once we have our infrastructure we will proceed to build a web application that receives messages and handles them properly to maintain availability.Then, we create a repository in DockerHub and upload our images. Finally, we will create an AWS EC2 virtual machine, install Docker/Docker-compose , and deploy our architecture in the cloud.

2 Objectives

1. To learn how containers work.
2. To understand the container's architecture.
3. To build a web Cloud architecture with AWS and Docker.
4. To identify the advantages of using containers.
5. To learn how to use Docker hub.

3 Definitions and Context

3.1 Docker

Docker is an open source platform that run applications and makes the process easier to develop, distribute. The applications that are built in the docker are packaged with all the supporting dependencies into a standard form called a container. These containers keep running in an isolated way on top of the operating system's kernel. [3]

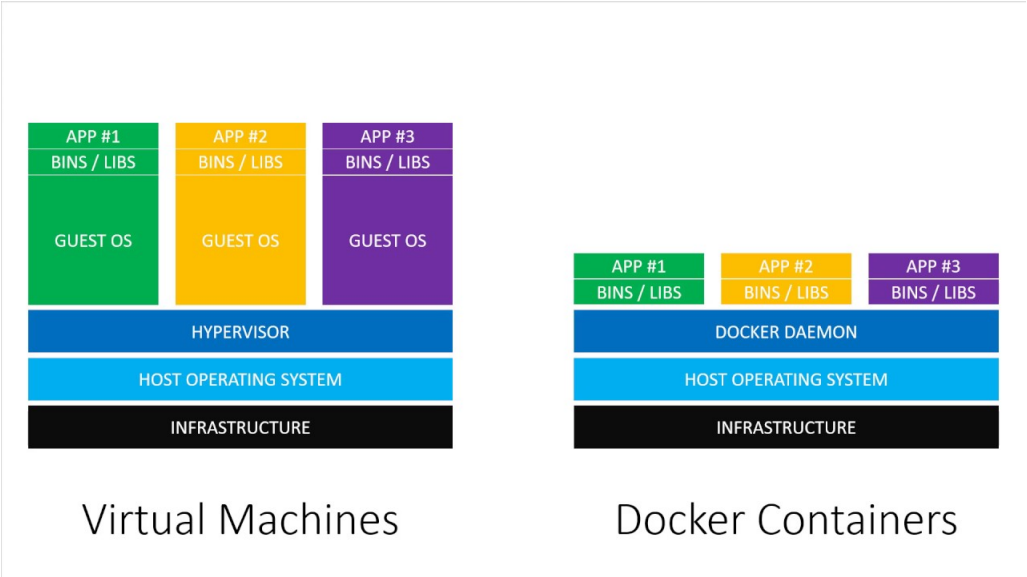

Figure 1: VM vs Container [3]

3.2 EC2

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) cloud. Using Amazon EC2 eliminates your need to invest in hardware up front, so you can develop and deploy applications faster. [1]

3.3 Load Balancing

Load balancing refers to efficiently distributing incoming network traffic across a group of backend servers, also known as a server farm or server pool.

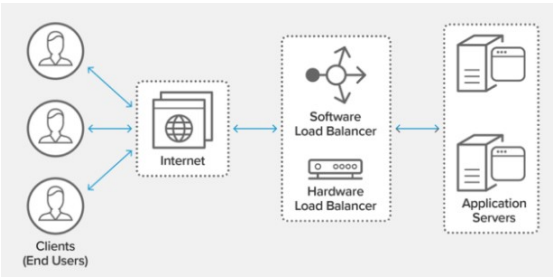

Figure 2: Load Balancing architecture [2]

4 Design of this project

4.1 Design's description:

The goal of this architecture is to create a load balancer that handles requests and distributes them equally among REST servers which manage the persistence layer by connecting to a database.

The load balancer sends a JSON to the REST servers and through the Round robin algorithm decides which REST server to send the request to, this server processes the json and inserts it into the database, as well as getting the last 10 messages from the database.

4.2 Architecture Diagrams

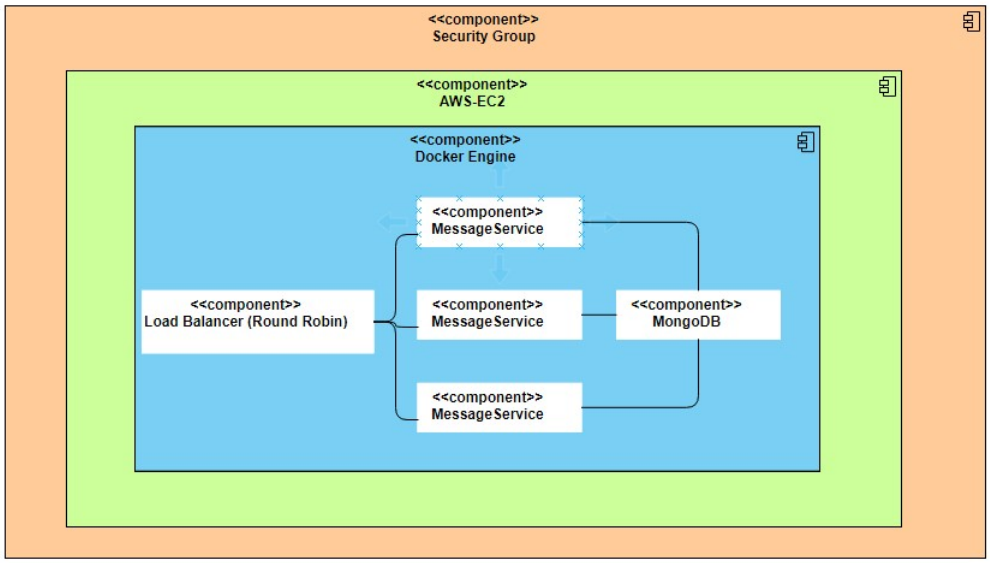

Figure 3: Architecture Diagram (Tech Stack)

4.3 Class diagrams:

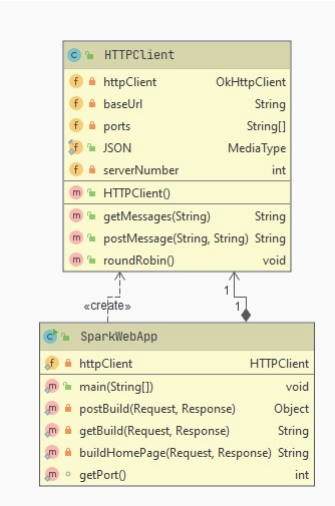

Figure 4: Class Diagram of the Load Balancer

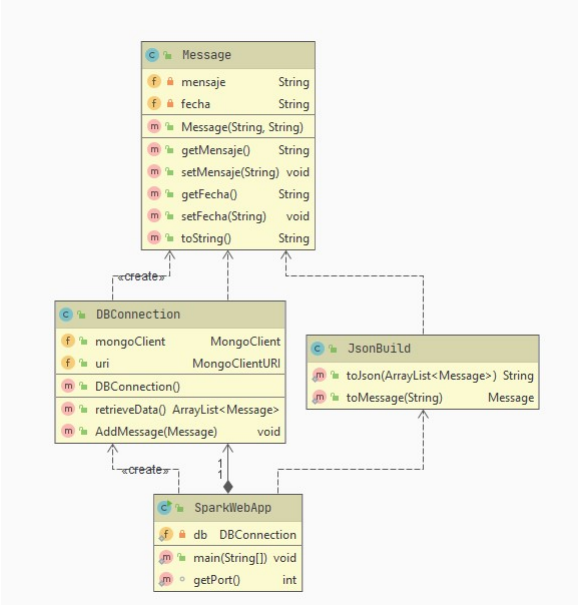

Figure 5: Class Diagram of REST node

5 Setting up this project at your local computer

Prerequisites

- Docker
- Git
- Docker-compose

1. Download the docker images

```
docker pull chan1100/lab5arep
docker pull chan1100/webdocker
```

2. Clone this repository:

```
git clone https://github.com/JohanS11/LAB5Arep.git
```

3. Go to the docker-compose directory

```
cd docker-compose
```

4. Execute the docker-compose.yml file

```
docker-compose up -d --scale web=3
```

5. Now you should be able to see this project at **http://ec2-3-82-154-139.compute-1.amazonaws.com:9001**

6 Conclusion

I have understood the importance of knowing how to use docker and the benefits of this new technology, it really corresponds to an impressive technological solution that serves for the development of applications, either in the cloud or on-premise

References

[1] What is amazon ec2? `https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html`. Accessed on 2020.

[2] What is load balancing? `https://www.nginx.com/resources/glossary/load-balancing/`. Accessed on 2020.

[3] Mohammad Ahmadi Babak Bashari Rad, Harrison John Bhatti. An introduction to docker and analysis of its performance. page 4, 2017.