



Modularización con virtualización e Introducción a Docker y a AWS

Autor:
Carlos Andrés Ramírez Torres

Arquitecturas Empresariales

Ingeniería de Sistemas

21 de Septiembre del 2020

1 Introducción

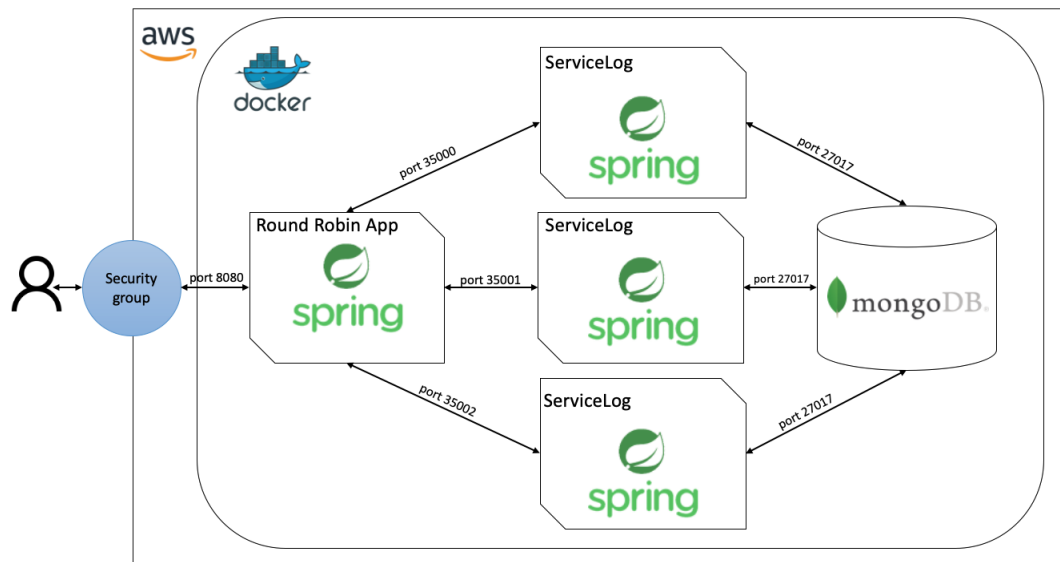
En este trabajo, se busca aplicar conocimientos de microservicios por medio del desarrollo de contenedores utilizando la herramienta Docker, adicionalmente la implementación de los contenedores en cloud utilizando el proveedor de servicios AWS, donde se hizo uso de una instancia ubuntu y la configuración apropiada de los security groups, para la conexión de los clientes por el puerto 8080 a un balanceador de carga Round Robin que es el encargado de dirigir las solicitudes a uno de los tres servicios disponibles para atenderlos, se utilizó spring boot como framework para otorgar servicios REST y comunicación entre aplicaciones utilizando rest template.

2 Conceptos

- Round Robin: El nombre de este algoritmo proviene del principio de turnos, donde cada persona obtiene una parte igual de algo por turnos. Es el algoritmo de programación más antiguo y simple, que se utiliza principalmente para realizar múltiples tareas. En la programación por turnos, cada tarea lista se ejecuta turno a turno solo en una cola cíclica durante un período de tiempo limitado.

3 Diseño

Los clientes de la aplicación acceden mediante la web utilizando la dirección asignada por la instancia EC2, por el puerto habilitado en el security group 8080, el que se conecta directamente con nuestra aplicación **round robin**, la cual contiene el frontend de la aplicación y un servicio **REST** encargado de administrar la carga de peticiones con el uso de **rest template** para distribuirlas por los 3 **servicios log** que se encuentran en docker con conexión interna a los puertos 35000, 35001 y 35002, los cuales realizan una petición a la base de datos por el puerto 27017 **mongoDB** para obtener los últimos 10 mensajes realizados por los usuarios, cabe afirmar que todos los servicios explicados anteriormente se encuentran en containers docker.



4 Conclusion

- Mediante el uso de rest template se pueden generar peticiones entre servicios rest.
- Los container permiten empaquetar aplicaciones con los recursos minimos necesarios para el funcionamiento y la portabilidad permite su uso en diferentes contextos como en el cloud como se puede observar en la explicación de la arquitectura.