



**Universidad de Guadalajara.**  
**Centro Universitario de Ciencias Exactas e Ingenierías.**



## **Ejercicio 10-2**

### **Microservicios con Docker**

### **Computación tolerante a fallas NRC 179961**

**Nombre del alumno:** Loredó Padilla Orlando Javier

**Código:** 217560328

**Departamento:** Departamento de ciencias computacionales

**Nombre del maestro:** López Franco Michel Emanuel

**Carrera:** Ingeniería en computación

**Sección:** D06

**Ciclo:** 23B

**Fecha:** 20/11/2023.

## Introducción

Ve el siguiente video y genera un reporte. (Nota: Profundiza en los temas buscando más información en internet.)

Mastering Chaos - A Netflix Guide to Microservices:

## Reporte

El video explica el funcionamiento de los microservicios de Netflix, al menos el como este los maneja para su plataforma. Ya sabemos que los microservicios sirven para hacer más ligero y práctico las funciones dentro de un programa o plataforma, se reparten las tareas en varios contenedores separados pero que trabajan a la parte, todo con el fin de que si llegase fallar alguno los demás no se vean afectados.

Una de las herramientas que se utilizan es la de ELB, o elastic load balancing; esta una herramienta que permite distribuir el trafico de red para que de esta manera se pueda mejorar la escalabilidad en una aplicación. Además, que podremos hacer un monitoreo del estado y rendimiento en donde lo estemos empleando, al igual que estaremos brindándole una mayor seguridad. Este se utiliza como el primer filtro de Netflix.

El siguiente es el ZUUL, que opera como enrutador y filtro, se encarga del direccionamiento de las acciones hechas en la página pudiendo de esta manera acceder a las distintas funciones que podemos usar. También tiene las funciones de autenticación, perspectivas, pruebas de estrés y canaria, el enrutamiento dinámico, migración de servicios, desconexión de carga, seguridad, entre otros.

Seguimos con el NCCP, o Netflix content control plane, el cual se encarga de dirigir a los usuarios a las ubicaciones especificadas, esto sirve para tener un mayor control la manera en como se nos direcciona hacia los servicios que hemos solicitado. Para esto se usa la CDN, esta se construye usando servidores personalizados llamados Open Connect Appliances (OCA), los cuales son servidores de caché.

El ultimo paso filtro es el API que se encarga de comunicar los equipos con los distintos servidores que este tiene disponible pasándole toda la información necesaria para operar. Estos cuatro elementos conforman la zona de Edge de los servicios de Netflix. Una vez se haya pasado por todas estas partes el usuario podrá hacer uso de los servicios que ofrece la plataforma ya que habremos entrado a la middle tier & platform donde se encontraran funciones mucho más específicas y que realizan menos trabajo o ya no operan como un filtro debido a que ya no son necesarios.

Cada parte esta replicada varias veces, la idea es que no exista un solo contenedor que realice una tarea si no que existan varios que realicen una misma tarea, de esta manera cuando llegue a fallar alguno el sistema siga funcionando sin afectar a los demás en lo que se intenta recuperar el contenedor que se haya caído y logrando así tener un sistema tolerante a fallas.

## Referencias

- S/A (S/F). Elastic Load Balancing en AWS. Recuperado de:  
[https://aws.amazon.com/es/elasticloadbalancing/#:~:text=Elastic%20Load%20Balancing%20\(ELB\)%20automatically,Gateway%20Load%20Balancer](https://aws.amazon.com/es/elasticloadbalancing/#:~:text=Elastic%20Load%20Balancing%20(ELB)%20automatically,Gateway%20Load%20Balancer)
- S/A (S/F). 8. Router and Filter: Zuul. Recuperado de:  
[https://cloud.spring.io/spring-cloud-netflix/multi/multi\\_router\\_and\\_filter\\_zuul.html#:~:text=Zuul%20is%20a%20JVM%2Dbased,side%20load%20balancer%20from%20Netflix](https://cloud.spring.io/spring-cloud-netflix/multi/multi_router_and_filter_zuul.html#:~:text=Zuul%20is%20a%20JVM%2Dbased,side%20load%20balancer%20from%20Netflix).