**KUBERNETES**

Antes de explicar que es **Kubernetes** hay que explicar otros conceptos, vamos a tomar **Docker** como ejemplo, para una aplicación que usa **Docker,** si quisiéramos ejecutar una app de **Node.js**, simplemente bastaría con escribir el comando “**docker run node.js**” y con esto ya tendríamos una app de **Node.js** pero, esta solo sería una instancia de la aplicación en un **host** **Docker**, pero supongamos que tu aplicación no es capaz de soportar la cantidad de peticiones que debe recibir, en ese caso, deberíamos **crear nuevos** **contenedores.** Por lo que ahora no tendrías un contenedor, si no dos y tendrías que estar pendiente de la cantidad de balance de carga de cada uno, y aparte de eso, tendrias que estar pendiente de que si alguno falla, ser capaz de detectarlo a tiempo y ejecutar los comandos apropiados para volver a ejecutar otra instancia de esa aplicación o que pasaria si el host que contiene el contenedor **Docker** falla? Entonces no se podría acceder a ninguno de los contenedores.

Es por estas razones y más **Kubernetes,**  --una tecnología que fue desarrollada por **Google--** resulta muy útil, funciona como un orquestador que nos permite automatizar la configuración, coordinacion y administracion de las aplicaciones, esto nos **permite desplegar cientos o miles de instancias de nuestra aplicación simultáneamente con un solo comando**, usando el kubernetes CLI y ejecutando el comando **“kubectl run --replicas=1000 my-web-server”** estaríamos ejecutando 1000 instancias de esa aplicación, también ofrece la posibilidad de configurarlos para que aumente o disminuya la cantidad de contenedores automáticamente, dependiendo de si hay mucho flujo de usuarios.

**Kubernetes** puede también actualizar un contenedor uno a la vez con un simple comando “**kubectl run rolling-update my-web-server --image-web-server:2”** o

también te da la posibilidad de volver a la versión anterior con el comando **“kubectl run rolling-update my-web-server --rollback”**, cabe acotar que este es un ejemplo por lo que tanto el nombre de la aplicación y la versión pueden cambiar en los comandos.

Estos comandos pueden ser aplicados a un porcentaje de tus contenedores para realizar diferentes pruebas.

**Kubernetes** se relaciona con **Docker** ya que usa host de **Docker** para hacer de **host en forma de contenedores Docker**, es como poner varias cajas(contenedores) dentro de una caja(contenedor) y que puedas administrar todas con una sola caja(contenedor)