



DIFERENCIAS ENTRE DOCKER, K8S, APACHE MESOS Y OPENSIFT

Computación Tolerante a Fallas

Murillo Cortes, Jeanette

Computación Tolerante a Fallas
Dr. Michel Emmanuel López Franco

Sección D06
Calendario 2022A

Diferencias entre Docker, K8s, Apache Mesos y OpenShift

Docker es permite que las aplicaciones se puedan desarrollar, ejecutar y desplegar mediante contenedores de software. Para crear uno, se deben compilar una serie de comandos que crea una imagen local o remotamente, creando un contenedor.

Kubernetes, es un orquestador de contenedores para implementar y administrarlos a gran escala, permitiendo el despliegue y la administración automática de aplicaciones, trabajos y servicios.

Apache Mesos gestiona los recursos del clúster mediante su monitorización, distribuyendo la carga entre los distintos servidores y evitando así sobrecargas.

OpenShift es una de las distribuciones de K8s. Sin embargo, es diferente porque mejora y añade ciertos aspectos de kubernetes, como en el aspecto de seguridad, donde OpenShift tiene políticas más estrictas. Además de que tiene características de escalabilidad e integración.

	Ventajas	Desventajas	Cómo funciona	Beneficios	Escalabilidad
Docker	<ul style="list-style-type: none"> Es una virtualización ligera. Permite almacenamiento en un repositorio. Las aplicaciones se almacenan mediante imágenes. Permite que la imagen se mueva a cualquier máquina. 	<ul style="list-style-type: none"> No es muy escalable cuando se habla de decenas, cientos o miles de contenedores (ocupa un orquestador). 	<ol style="list-style-type: none"> En P00, la imagen es la clase y los contenedores son los objetos. Se crea una imagen, así empaqueta la información con todo su entorno de todas las librerías y archivos que necesita. 	<ul style="list-style-type: none"> Administración aislada de cada imagen. Bajos costos. Puede crear tareas programadas o batch. Dan soporte a nivel de aplicación. 	No es muy escalable cuando se requieren muchos contenedores.
Kubernetes (K8s) <small>Su nombre viene del Griego y hace referencia a un timón.</small>	<ul style="list-style-type: none"> Orquesta contenedores. Está basado en Borg, el cuál busca revivir los contenedores que contiene. Puede manejar muchos contenedores a través de muchos nodos. 	<ul style="list-style-type: none"> Es ligero. No se asegura que sobrevivan los contenedores. 	<ol style="list-style-type: none"> Se crea mediante un deployment (el cuál es un template para crear pods). Después, el controller manager se encarga de crear esos pods en el clúster. 	<ul style="list-style-type: none"> Puede hacer uso extensivo de servicios y microservicios. Dan soporte a nivel de aplicación. Crea replicas de la aplicación. 	Se puede escalar a gran escala.
Apache Mesos	<ul style="list-style-type: none"> Permite utilizar el clúster para correr los frameworks sobre los cuales se ejecuta la 	<ul style="list-style-type: none"> Es ligero A veces en seguridad tiene políticas 	<ol style="list-style-type: none"> utiliza un sistema de dos niveles de planificación donde decide cuántos 	<ul style="list-style-type: none"> Corre grandes aplicaciones que tienen volúmenes altos de 	Tiene auto escalado lineal: puede escalar

	<p>aplicación, distribuyendo la carga entre los distintos servidores y evitando así sobrecargas.</p> <ul style="list-style-type: none"> Logra un óptimo desempeño. 	bastante estrictas	<p>recursos asignar a cada Framework y el Framework decide qué recursos aceptar y qué tareas correr en esos recursos.</p>	<p>datos y es fácilment e de forma lineal hasta 10.000 nodos.</p> <p>– Destaca su tolerancia a fallos.</p>	
OpenShift	<ul style="list-style-type: none"> Implementa mayor seguridad. Tiene etiquetas de estados. Gestión más sencilla para desarrolladores. Ofrece una API 	<ul style="list-style-type: none"> A veces en seguridad tiene políticas bastante estrictas. En una versión gratuita, su durabilidad es muy pequeña, como de 24 horas. 	<ol style="list-style-type: none"> El proyecto se guarda en un repositorio de GitHub, para que al desplegarse lo lea OpenShift. El código leído se inyecta a una imagen de Docker de forma automática. 	<p>– Puede construir imágenes</p> <p>– Puede construir imágenes basadas en repositorios de GitHub.</p>	<p>Es escalable, pero su escalabilidad también depende de nuestros recursos, además de que se paga de acuerdo con los mismos.</p>

DESARROLLO DE LA PRÁCTICA - PREINSTALACIONES

Para el desarrollo de la práctica se hizo uso de:

- Rancher
- K3D
- Chocolatey

INSTALACIÓN DE RANCHER

Para descargarlo se debe de ir a la página de Rancherdesktop.io en el apartado de instalación, link: <https://docs.rancherdesktop.io/getting-started/installation/>.

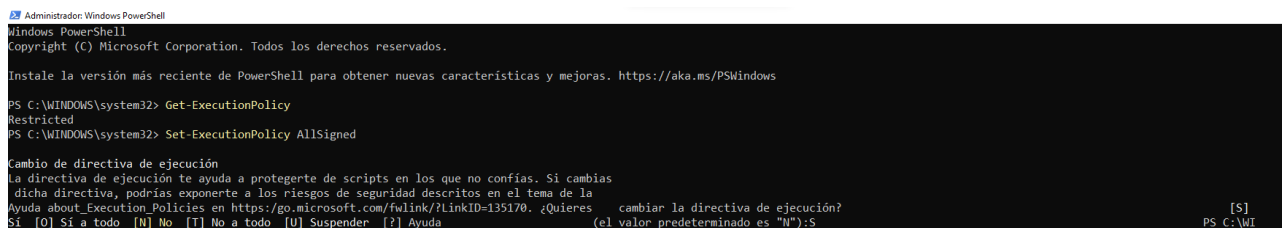
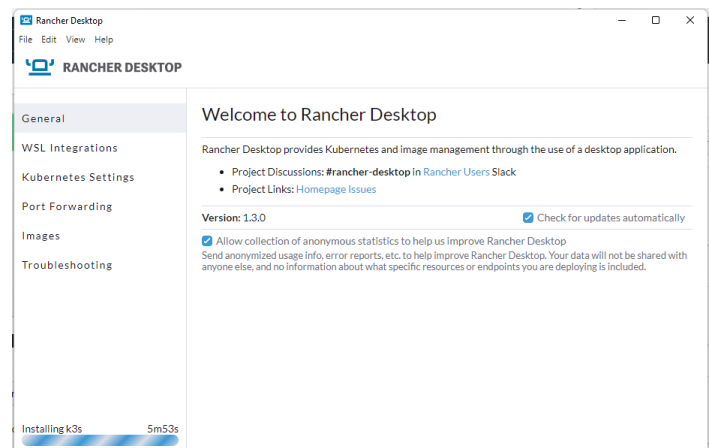
De manera que una vez ahí, se debe de ir al apartado de GitHub y descargar la opción para Windows la cuál corresponde al .exe.

De esta manera podremos ejecutarlo, de forma que la primera vez que se abra nos preguntará si queremos que se instalen instancias para kubernetes, a lo cuál daremos que si. Esto va a causar que se observe la descarga en el lado izquierdo de la aplicación. Una vez terminado lo anterior, se va a poder abrir la aplicación para comenzar a utilizarla y administrar el clúster.

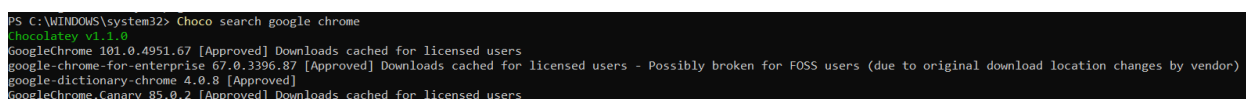
DESARROLLO DE LA PRÁCTICA

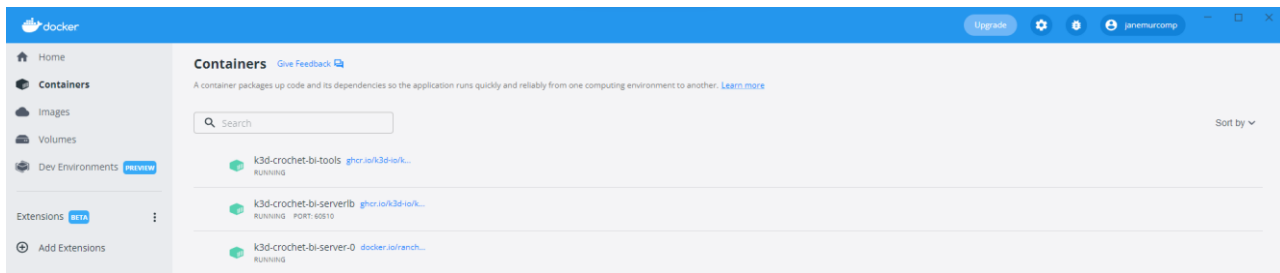
En el desarrollo de la práctica, para el correcto funcionamiento de K3S con Rancher, se hizo uso de K3D, el cuál nos va a permitir gestionar el clúster que se creó durante la práctica. De esta manera, primeramente, se utilizaron los comandos `Get-ExecutionPolicy` y `Set-ExecutionPolicy AllSigned` para la configuración de la

directiva de ejecución de Windows PowerShell, además del comando `Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))` para poder hacer uso de Chocolatey mediante PowerShell abierto como administrador, como se muestra en la siguiente imagen.



Además, se comprobó que Chocolatey si estuviera instalado correctamente, para de esta manera utilizar los comandos `Choco search google Chrome`.



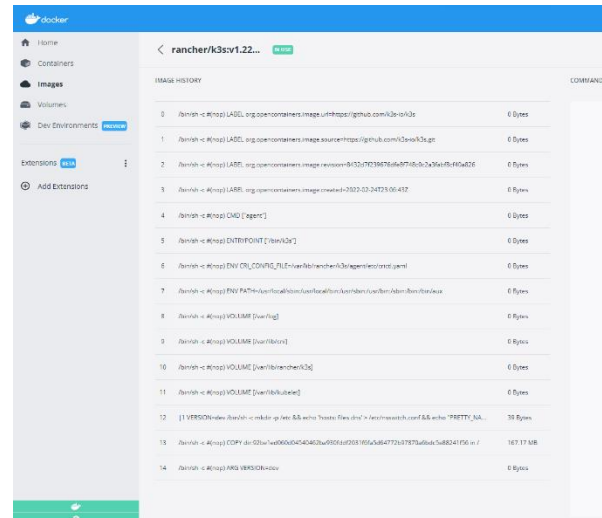


Una vez creada la imagen de Docker, se implementó el archivo .yaml que contenía una configuración definida para la aplicación, de manera que para aplicar y correr dicho archivo, se hizo uso del comando `kubectl apply -f crochet-bi-depl.yaml` para el deployment.

```
PS C:\Users\patoa\Documents\Escuela\Kubernetes-docker-minikube\yaml> kubectl apply -f crochet-bi-depl.yaml
deployment.apps/crochet-bi-depl created
```

De manera que mediante `kubectl get pods` se pueden ver los pods y `kubectl get services` para ver los servicios que configuramos.

```
PS C:\Users\patoa\Documents\Escuela\Kubernetes-docker-minikube\yaml> kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
crochet-bi-depl-f59b67dc8-4crw8     0/1     CrashLoopBackOff   4 (69s ago)    2m53s
crochet-bi-depl-f59b67dc8-79wd9     0/1     CrashLoopBackOff   4 (79s ago)    2m53s
crochet-bi-depl-f59b67dc8-qkpfz     0/1     CrashLoopBackOff   4 (77s ago)    2m53s
PS C:\Users\patoa\Documents\Escuela\Kubernetes-docker-minikube\yaml> kubectl get services
NAME      TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes ClusterIP   10.96.0.1     <none>        443/TCP    24m
PS C:\Users\patoa\Documents\Escuela\Kubernetes-docker-minikube\yaml> kubectl get pods
```



CONCLUSIONES

El desarrollo de esta práctica me permitió conocer más a fondo acerca del funcionamiento de todas estas nuevas tecnologías que utilizan, permiten desarrollar y administrar contenedores, de manera que esto nos brinda la posibilidad de que en un futuro podamos implementarlo en nuestros programas para que estos tengan mayor escalabilidad, uso de servicios y mejor desarrollo en cuanto costos. Por su parte, también me permitió ver cómo es que nuestras aplicaciones se pueden hacer tolerantes a fallos mediante las mismas.

BIBLIOGRAFÍA

- Arias, Jesús. *"Memoria_rancher_jesus_arias.pdf"* (gonzalonazareno.org). Recuperado del URL: https://dit.gonzalonazareno.org/gestiona/proyectos/2018-19/memoria_rancher_jesus_arias.pdf
- *"Cómo instalar Chocolatey en Windows 11 usando PowerShell - Howto88"*. Recuperado del URL: <https://howto88.com/es/como-instalar-chocolatey-en-windows-11-usando-powershell>
- Fernández, Alberto. (2021). *"Rancher, un solo cluster para gobernarlos a todos - Paradigma"* (paradigmadigital.com) <https://www.paradigmadigital.com/dev/rancher-cluster-para-gobernarlos-a-todos/>
- *"Installing chocolatey on windows 11 Code Example"* (codegrepper.com). Recuperado del URL: <https://www.codegrepper.com/code-examples/shell/installing+chocolatey+on+windows+11>
- *"¿QUE ES KUBERNETES? - Introducción al orquestador más usado"*. Recuperado del URL: <https://www.youtube.com/watch?v=0Tf0KxK1QNo&t=377s>