

PRACTICA 5 – CREANDO IMÁGENES

El objetivo de esta actividad es crear una nueva imagen a partir de una imagen dada y publicarla en el registro docker hub.

Descarga la imagen de alpine. Lanzala en un contenedor con los siguientes parámetros

```
docker run --privileged --network=host -it --name node alpine /bin/sh
```

- `--privileged`: Esto le otorga al contenedor acceso completo a todos los dispositivos en el host, lo cual incluye el kernel. Es necesario para acceder al kernel desde el contenedor.
- `--network=host`: Esto configura el contenedor para usar el modo de red host, lo que permite al contenedor compartir el espacio de red del host. Esto significa que el contenedor usará la red del host directamente.

Una vez dentro actualiza el instalador de paquetes apk

```
apk update
apk add net-tools
apk add iputils-ping
```

Ahora haz un ping desde el host principal y verás que es visible el contenedor. Hemos creado una red a la cual pertenecen tanto el host como el contenedor.

Instala nodejs

```
apk add nodejs
```

Comprueba que funciona

```
/ # node -v
v20.11.1
```

Instala ahora nano para editar un fichero js

```
apk add nano
```





y crea el fichero /home/uno.js con el siguiente contenido

```
console.log("Hola Mundo");
```

finalmente, ejecútalo

```
/ # node /home/uno.js  
Hola Mundo
```

La imagen estaba lanzada en el contenedor denominado "contenedor"

<input type="checkbox"/>	Name	Image	Status	CPU (%)	Actions
<input type="checkbox"/>	 contenedor b4b7350447fb	alpine	Exited (137)	N	  

O bien, desde el CLI con

```
docker ps -a  
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES  
b4b7350447fb   alpine    "/bin/sh" 26 minutes ago Exited (137) 8 minutes ago  
contenedor
```

Ahora vamos a crear una nueva imagen partiendo de la anterior pero incluyendo las modificaciones hechas.

```
docker commit contenedor peperodri/nueva (debes poner el nombre de usuario  
creado en docker hub)
```

Podemos ver que ya se ha creado la nueva imagen

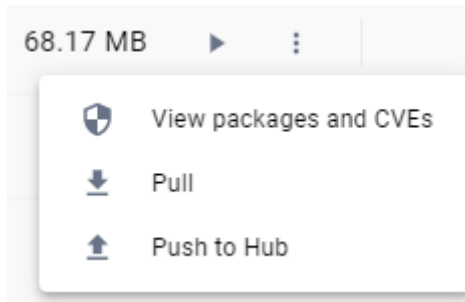
```
PS C:\Users\profesor> docker image ls -a  
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE  
peperodri/nueva      tag       cda4f4d2688e   9 minutes ago 68.2MB  
alpine          latest    05455a08881e   3 weeks ago   7.38MB
```

Podrás comprobar que esta imagen nueva ocupa mucho más ya que hemos añadido algunos módulos nuevos.

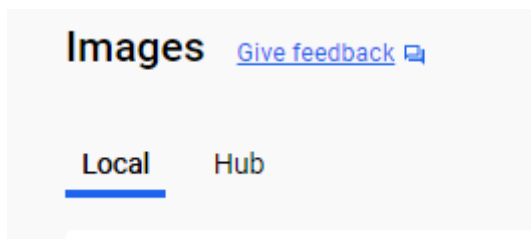
El siguiente paso es publicarla en docker hub. Para ello tendremos que loguearnos con las credenciales válidas

```
PS C:\Users\profesor> docker login
Authenticating with existing credentials...
Login Succeeded
```







Ahora, en la imagen que queremos publicar elegimos la opción push to hub



Si nos vamos a la pestaña hub



Tendremos

	Tags	OS	Vulnerabilities	Last pu
 peperodri/nueva	tag		 Inactive	5 minu
 peperodri/busybox	latest		 Inactive	2 years