Docker es una de las tecnologías más de moda y poco a poco entrará en prácticamente todas las plataformas. Ahora bien ¿qué es ? y ¿cómo funciona?. Docker esta relacionado con el mundo del despliegue de aplicaciones. Todos desarrollamos aplicaciones y las instalamos en nuestros servidores .

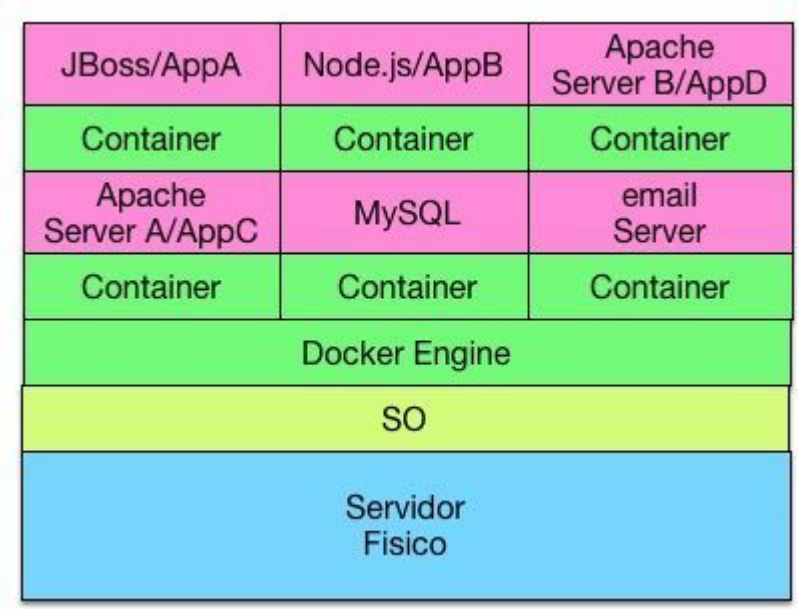


Con el paso de los años ha dejado de ser habitual trabajar de esta forma y se ha optado por soluciones de virtualización tipo VMWare o Virtual Box .Estas permiten aprovechar mejor los recursos de nuestro hardware. Podemos disponer de una máquina de doble procesador físico con 16 núcleos y 64 GB de RAM y virtualizar varios sistemas operativos de forma simultánea . En cada sistema operativo instalaremos lo que necesitemos MySQL , Tomcat , Apache , JBoss etc.



Esta solución es mejor que la anterior en la que utilizabamos servidores físicos y nos

permite una gestión mucho más moderna de los recursos . La pregunta más importante es : ¿Es esta solución óptima? . Si revisamos el esquema nos daremos cuenta que cada máquina virtual contiene un sistema operativo COMPLETO . ¿Es esto necesario? . Es decir, si nosotros en un sistema operativo lo único que necesitamos es instalar un MySQL , ¿Es obligatorio instalar el sistema operativo completo?. O podemos instalar un núcleo mínimo sobre el que desplegar MySQL. Esta es una idea muy interesante y es el concepto sobre el que Docker se basa. Docker genera CONTENEDORES . Estos contenedores se encargan de almacenar un sistema operativo mínimo y la aplicación que deseamos desplegar.



De esta manera podemos sacar un mayor rendimiento al hardware del que disponemos ya que a partir de este momento no es necesario virtualizar el sistema operativo entero sino que podemos crear contenedores con nuestras necesidades especificas. De esta forma aumentamos la capacidad de despliegue de nuestras máquinas físicas y ahorramos costes.

Un ejemplo de Docker

Utilizar docker no es complicado , aunque si es cierto que todavía la información es un poco confusa. El primer paso es instalarlo desde la página web. Una vez instalado empezar es

relativamente fácil. Abrimos una linea de comandos y escribimos

docker search apache2

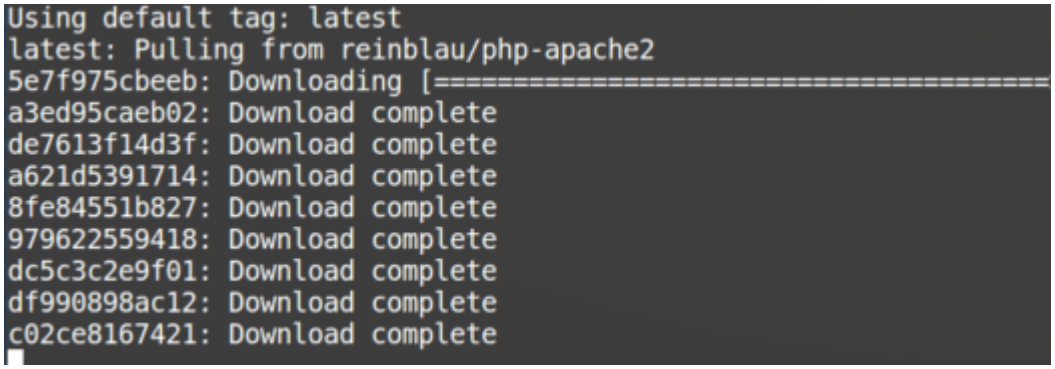
Este comando hace que docker se conecte a su repositorio y nos busque un contenedor de apache2



El siguiente paso es seleccionar un contenedor e instalarlo , en este caso podemos usar el servidor Apache2 para PHP. Escribiremos en linea de comandos

docker pull reinblau/php-apache2

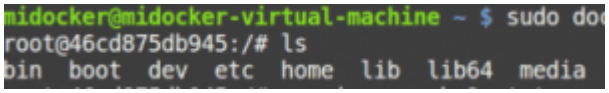
Se descargará la imagen del contenedor



Descargada la imagen del contenedor nos queda arrancarlo y crear una instancia para poder trabajar con ella. Para ello usamos el comando run.

docker run -it reinblau/php-apache2 /bin/bash

Ya tenemos un contenedor arrancado y nos hemos conectado a su shell.



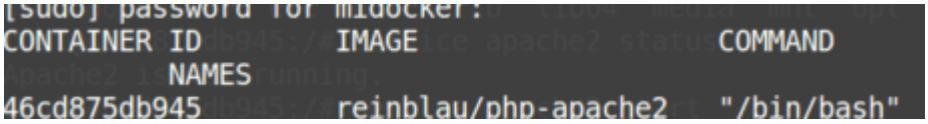
El siguiente paso es escribir desde la shell :

service apache2 start

Esto arrancará apache2 desde el contenedor y asignará una ip propia al contenedor . Para saber la ip del contenedor abriremos otra shell y escribiremos

docker ps -a

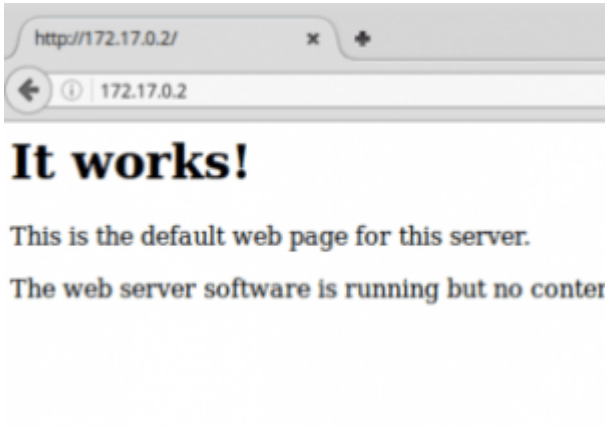
Que nos devolverá los contenedores existentes y sus ids



Con la id del contenedor solicitaremos inspeccionar el contenedor

docker inspect 46cd875db945

Esto devuelve una estructura JSON con mucha información entre ella la IP del container. Abrimos un navegador y accedemos a la IP.



Tenemos nuestro primer contenedor de Docker arrancado con un Apache. Estas tecnologías tienen un presente y futuro brillante. La llegada de las arquitecturas orientadas a microservicios les darán un plus. Habrá situaciones en las que encaje mejor un servidor físico , otras un contenedor y otras una máquina virtual. Lo importante es tener opciones.