

CONOCIENDO A LA PLATAFORMA DOCKER

Lic. Grover Magueño Gordillo
Carrera de Ingeniería de Sistemas, Escuela Militar de Ingeniera
La Paz, Bolivia
grovemilton@gmail.com.

KNOWING THE DOCKER PLATFORM

Resumen— El presente trabajo pretende dar a conocer DOCKER la cual es una plataforma de portabilidad de aplicaciones de software mediante la creación y uso de imágenes de contenedores.

Palabras Claves— Contenedor, Imagen, Virtualización, Sistema Operativo, Nucleo.

Abstract - The present work pretends to present DOCKER which is a platform of portability of software applications through the creation and use of container images.

Keywords - Container, Image, Virtualization, Operating System, Nucleus

I. INTRODUCCIÓN.

Seguramente en muchas ocasiones has tenido que emprender un proyecto en el desarrollo de software en una plataforma “X” y has tenido la necesidad de que ese producto de software pueda ejecutarse en otra plataforma “Y”, lo que te llevaría quizá a reescribir el código de manera parcial o de manera total; la portabilidad de productos de software ha sido un dolor de cabeza para los desarrolladores, pero en los últimos años esta figura está cambiando gracias a plataformas como DOCKER.

II. ¿QUE ES DOCKER?

DOCKER es una tecnología de código abierto, cuyo objetivo es brindar la portabilidad de aplicaciones entre distintas plataformas a través de la virtualización de contenedores a nivel de sistema operativo.

DOCKER básicamente crea contenedores ligeros y portables para las aplicaciones software con el objetivo de que estas puedan ejecutarse en cualquier máquina en la que esté instalado DOCKER independientemente del sistema operativo.

En resumen, DOCKER nos permite el despliegue de aplicaciones mediante el uso de contenedores que albergan todas las rutinas y librerías que la aplicación necesita para funcionar de manera independiente y aislada.

III. COMO FUNCIONA DOCKER

Antes de Describir cual es el funcionamiento de DOCKER debemos tener algunos conceptos claros.

- Contenedor.

De manera abstracta un contenedor para DOCKER es muy similar a un contenedor de carga, el cual se enmarca en un estándar en cuanto a dimensiones y formas, entonces un contenedor de DOCKER servirá para transportar la aplicación y todo lo que necesita para funcionar.

- **Imagen.**
En informática una imagen es un archivo que contiene una copia fiel de todo un sistema de archivos.
- **Virtualización.**
Es la creación a través de software de una versión virtual de algún recurso informático, como un recurso hardware, un sistema operativo, un HDD, ODD u otros.

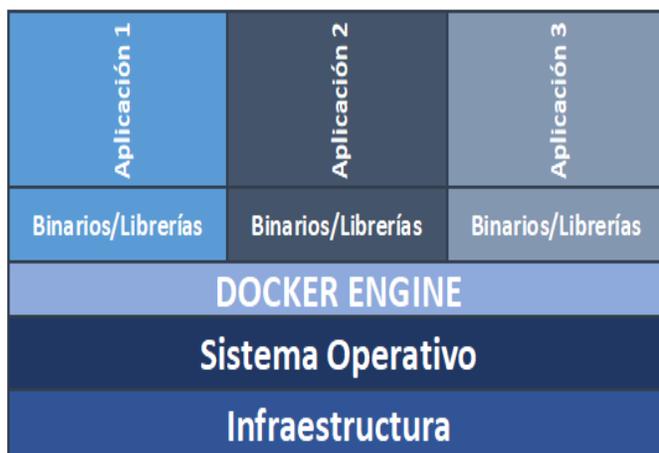
DOCKER se despliega bajo un núcleo GNU/LINUX, aunque actualmente existen implementaciones para Windows y MAC, en principio se trabaja sobre una imagen base, en la cual realizaremos cambios según nuestras necesidades.

Una vez culminada con esta tarea y confirmar estos cambios crearemos una nueva imagen con DOCKER.

Esta nueva imagen contenedor contiene solo los recursos necesarios para que funcione nuestra aplicación (Servidores de bases de datos, servidores web, lenguajes de programación, etc.), reduciendo drásticamente su volumen en cuanto tamaño.

La Imagen 1 muestra la estructura de

Imagen 1: Estructura Docker



Fuente: Elaboración Propia

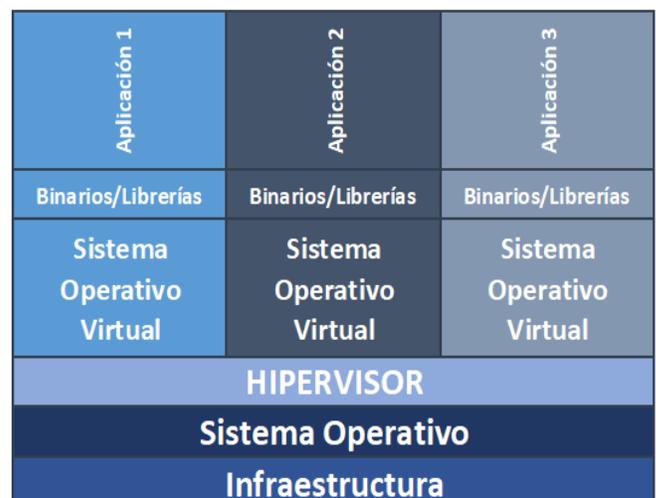
DOCKER.

En base a esta estructura DOCKER desplegará cada contenedor sobre DOCKER ENGINE que se encargará de

copiar la base, la imagen y las diferentes capas para darnos el entorno que queremos desplegar (Servidores de bases de datos, servidores web, lenguajes de programación, etc.), para empezar a trabajar.

Resumiendo, DOCKER levanta una imagen de un contenedor el cual tiene solo los recursos (Binarios y librerías) necesarios para que una determinada aplicación se ejecute, cosa muy distinta a un sistema de virtualización que por lo general carga todo un sistema operativo y sobre este recién las aplicaciones que nos interesan ejecutar, véase la imagen 2.

Imagen 2: Arquitectura de virtualización



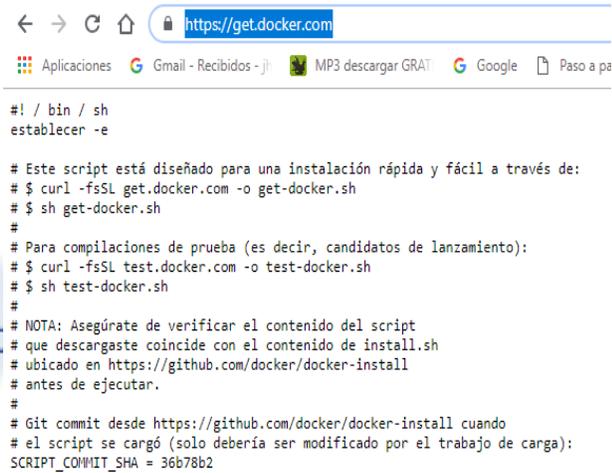
Fuente: Elaboración Propia

IV. CARACTERÍSTICAS DE DOCKER

DOCKER presenta las siguientes características.

- Auto gestión de contenedores.
- Fiabilidad.
- Despliegue de múltiples contenedores en un mismo anfitrión.
- Aplicaciones libres de dependencias.
- Contenedores muy livianos.
- Facilidad en el transporte y despliegue de contenedores.
- Compatibilidad en Multi-Sistema.
- Disponibilidad de contenedores en el repositorio DOCKER.

Imagen 3: Script de instalación



```

#!/bin/sh
establecer -e

# Este script está diseñado para una instalación rápida y fácil a través de:
# $ curl -fsSL get.docker.com -o get-docker.sh
# $ sh get-docker.sh
#
# Para compilaciones de prueba (es decir, candidatos de lanzamiento):
# $ curl -fsSL test.docker.com -o test-docker.sh
# $ sh test-docker.sh
#
# NOTA: Asegúrate de verificar el contenido del script
# que descargaste coincide con el contenido de install.sh
# ubicado en https://github.com/docker/docker-install
# antes de ejecutar.
#
# Git commit desde https://github.com/docker/docker-install cuando
# el script se cargó (solo debería ser modificado por el trabajo de carga):
SCRIPT_COMMIT_SHA = 36b78b2
    
```

Fuente: Elaboración Propia

V. COMO EMPEZAR A TRABAJAR CON DOCKER.

Para empezar a trabajar con DOCKER lo primero que debemos hacer es instalarlo entrando a la siguiente página web <https://get.docker.com/> donde nos desplegará el script que se muestra en la imagen 3.

Ubique la siguiente línea “**curl -fsSL get.docker.com -o get-docker.sh**” que se encuentra en la página mencionada anteriormente y cópielo a su terminal y ejecútela, inmediatamente procederá a instalar DOCKER.



```
$ sudo curl -sSL https://get.docker.com/ | sh
```

Una vez instalado DOCKER acceda a la página oficial en <https://www.docker.com/> donde podrá acceder a recursos y soporte para empezará a trabajar con esta plataforma.

VI. CONCLUSIONES

DOCKER es una plataforma que nos permite crear contenedores de propósito específico, portables y ligeros, lo que nos facilita migrar de un entorno de trabajo a otro sin necesidad de volver a instalar todo lo que necesitamos para trabajar en desarrollo u otros rubros informáticos.

BIBLIOGRAFÍA

Cabrera Gomes de la Torre, m. (s.f.). Gestión de contenedores docker*kubernetes.
 Fernandez Gines, X. (2018). Seguridad en Docker.
 Hdez ALM, U. (29 de 06 de 2016). Docker, manual básico en español (2017). Obtenido de <https://rootsudo.wordpress.com/2016/06/29/docker-manual-en-espanol/>
 Rodriguez Gayoso, R. (2017). Recetas Docker Documentation.

RECEPCION: 18/02//2019
 APROBACION: 21/03/2019

