

Resumen – Docker

Anel Atencio

8-950-868

Docker es un proyecto de código abierto para automatizar la implementación de aplicaciones, nos permite ejecutar un proceso de manera aislada de los procesos de la máquina y define todas las dependencias que necesita para ejecutarse. Si se necesita ejecutar una aplicación que solo funciona en una distribución de Linux en específico con Docker se pueden empaquetar en una imagen para que toda la información esté disponible a la hora de ejecutar el programa sin la necesidad de tener que depender del sistema operativo principal de la máquina.

Docker facilita el proceso de creación y diseño de contenedores, el envío de imágenes y creación de versiones de imágenes. Los contenedores se usan como máquinas virtuales pero a diferencia suelen usarse para levantar máquinas independientes con SO más ligeros. Docker permite crear, implementar, copiar y mover los contenedores de un entorno a otro, como los contenedores requieren menos recursos se inician rápidamente y son fáciles de implementar esto significa que se pueden ejecutar más servicios en una máquina. Docker es un runtime para contenedores y trabajan con imágenes que son plantillas de solo lectura que definen su contenedor, esta imagen contiene el código que se ejecutará incluido cualquier dependencia que el código necesite.

Los principales beneficios de los contenedores es que son más ligeros que una máquina virtual, no necesitan de instalaciones de sistemas operativos, utiliza menor recursos de la máquina, permite que haya muchos contenedores por equipo físico y mejora la portabilidad de aplicaciones.

¿Cómo justificaría la implementación de Docker para una empresa?

Si quisiera implementar Docker en una empresa, primero vería de qué trata esta empresa porque Docker funciona para empresas que utilizan aplicaciones en sistemas operativos de distribución Linux por lo que vería si ese es el caso. Luego, vería si están utilizando virtual box, u otro servicio para máquinas virtuales, les diría que Docker es mejor porque es más ligero debido a que no tiene la necesidad de instalar un nuevo sistema operativo para ejecutar una aplicación, en cualquier computador con UNIX podría ejecutarla. Les diría que esto es posible porque Docker usa “contenedores” que serían como las máquinas virtuales pero con imágenes portables que se podrían ejecutar en cualquier computador y, al ser portable y más ligero, podría haber muchas imágenes diferentes para distintas aplicaciones en una misma máquina, sin la necesidad de tener que consumir más equipo físico para estas aplicaciones. Por lo que Docker sería una solución viable para que la empresa pueda ahorrar costos en equipos físicos y tener todas las aplicaciones más centralizadas y disponibles para cualquiera sin consumir muchos recursos.

PRUEBA EN DOCKER

Ejecute una sola tarea en un contenedor de Alpine Linux

```
[node1] (local) root@192.168.0.18 ~
$ docker container run alpine hostname
ef358b2c8102
[node1] (local) root@192.168.0.18 ~
$
```

Listar todos los contenedores

```
8a8ec78ac87c
[node1] (local) root@192.168.0.18 ~
$ docker container ls --all
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS
PORTS              NAMES
8a8ec78ac87c        alpine             "hostname"          3 seconds ago       Exited (0) 2 seconds a
go                  eloquent_galois
b98651b5fc5b        alpine             "hostname"          20 seconds ago       Exited (0) 19 seconds
ago                 youthful_ptolemy
```

Ejecute un contenedor Docker

```
[node1] (local) root@192.168.0.18 ~
$ docker container run --interactive --tty --rm ubuntu bash
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
6a5697faee43: Pull complete
ba13d3bc422b: Pull complete
a254829d9e55: Pull complete
Digest: sha256:fff16eeala8ae92867721d90c59a75652ea66d29c05294e6e2f898704bdb8cf1
Status: Downloaded newer image for ubuntu:latest
```

Comandos de Linux dentro del contenedor

```
root@7f688bda0771:/# ls /
bin  dev  home  lib32  libx32  mnt  proc  run  srv  tmp  var
boot  etc  lib  lib64  media  opt  root  sbin  sys  usr
root@7f688bda0771:/# ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1   0.6   0.0   4108   3576 pts/0    Ss   20:52   0:00 bash
root        10   0.0   0.0   5888   2868 pts/0    R+   20:52   0:00 ps aux
root@7f688bda0771:/# cat /etc/issue
Ubuntu 20.04.1 LTS \n \l
```

Verificación de la versión de la máquina virtual

```
[nodo1] (local) root@192.168.0.18 ~  
$ cat / etc / problema  
Bienvenido a Alpine Linux 3.12  
Kernel \ r en una \ m (\ l)
```

En esta prueba realizada en: https://training.play-with-docker.com/beginner-linux/#Task_0. Se pudo aprender cómo se utiliza Docker mediante la terminal para crear y utilizar los contenedores.