

Práctica 5. Creación de contenedores Docker

Ejercicio 1.

Crear un contenedor personalizado. Exponer los puertos para que sean accesibles desde la red Zerotier. No es necesario instalar Zerotier en el contenedor docker, sólo en la máquina principal (nuestro nodo cpdX). Utilizar cualquier otro servicio distinto a NGINX.

Para la creación del contenedor personalizado se ha deseado hacer uso de un Dockerfile para que toda la creación se realizase de forma automática. El contenido de este Dockerfile se muestra a continuación:

FROM ubuntu

Instala los distintos paquetes para python3

*RUN *

*apt-get update && *

*apt-get install -y git-all && *

*apt-get install -y apache2 && *

se instalan los paquetes de python con pip

*apt-get install -y python3-pip && *

se crea repositorio de trabajo

mkdir experiments

```

# Define variables de entorno

ENV APACHE_RUN_USER www-data

ENV APACHE_RUN_GROUP www-data

ENV APACHE_PID_FILE /var/run/apache2.pid

ENV APACHE_RUN_DIR /var/run/apache2

ENV APACHE_LOCK_DIR /var/lock/apache2

ENV APACHE_LOG_DIR /var/log/apache2

# Directorios que pueden montarse

VOLUME ["/var/www/html"]

# Define directorio de trabajo

WORKDIR "experiments"

# Define orden por defecto

CMD ["apache2", "-D", "FOREGROUND"]

# Puertos que se exponen exteriormente

EXPOSE 80

EXPOSE 443

```

Para la creación de este contenedor a partir del Dockerfile se usa el siguiente comando:

```
docker build --rm -t datascience .
```

Ahora, una vez que se ha instalado el contenedor, lo corremos conectando el puerto 80 de nuestra máquina al del contenedor y además, vemos cuál es la IP de la red zero-tier para poder acceder desde la máquina local:

```
docker run -p80:80 datascience
```

```
zerotier-cli join id_nodo
```

```
root@c4:~# docker run -p80:80 datascience
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message

UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:349022 errors:0 dropped:0 overruns:0 frame:0
TX packets:340411 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:48428950 (48.4 MB)  TX bytes:59225634 (59.2 MB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:39614 errors:0 dropped:0 overruns:0 frame:0
          TX packets:39614 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:12036344 (12.0 MB)  TX bytes:12036344 (12.0 MB)

veth8c7c67e Link encap:Ethernet  HWaddr ce:ee:1a:8d:b6:64
          inet6 addr: fe80::ccee:1aff:fe8d:b664/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:8 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3685 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:648 (648.0 B)  TX bytes:155082 (155.0 KB)

zt2      Link encap:Ethernet  HWaddr f6:8e:8a:a6:68:5f
          inet addr:192.168.193.106  Bcast:192.168.193.255  Mask:255.255.255
          inet6 addr: fe80::f48e:8aff:fea6:685f/64 Scope:Link
          inet6 addr: fc7b:296d:b4ea:74ca:2b56::1/40 Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:2800  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:946 (946.0 B)

root@c4:~#
```

Y ahora podemos acceder a la dirección IP que se puede observar en la imagen superior, y entramos a la página configurada por defecto de Apache:



