Centro de procesamiento de datos Francisco Carrillo Pérez 23 de noviembre de 2017

## 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

 $RUN \setminus$ 

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

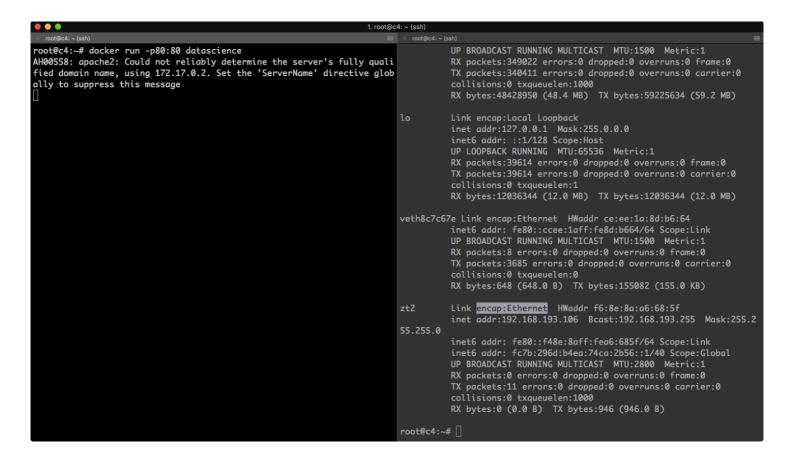
Para la creación de este contenedor a partir del Dockerfile se usa el siguiente comando:

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



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:

