Introducción a Docker - Parte 1

@javierprovecho

¿Que es Docker?

- Plataforma abierta para crear y ejecutar aplicaciones distribuidas.
- Destinado a desarrolladores y administradores.
- Construcción basada en componentes separados.
- Ejecución en distintas plataformas (servers, desktops, laptops…) con una misma compilación.

¿En que esta basado? (requisitos)

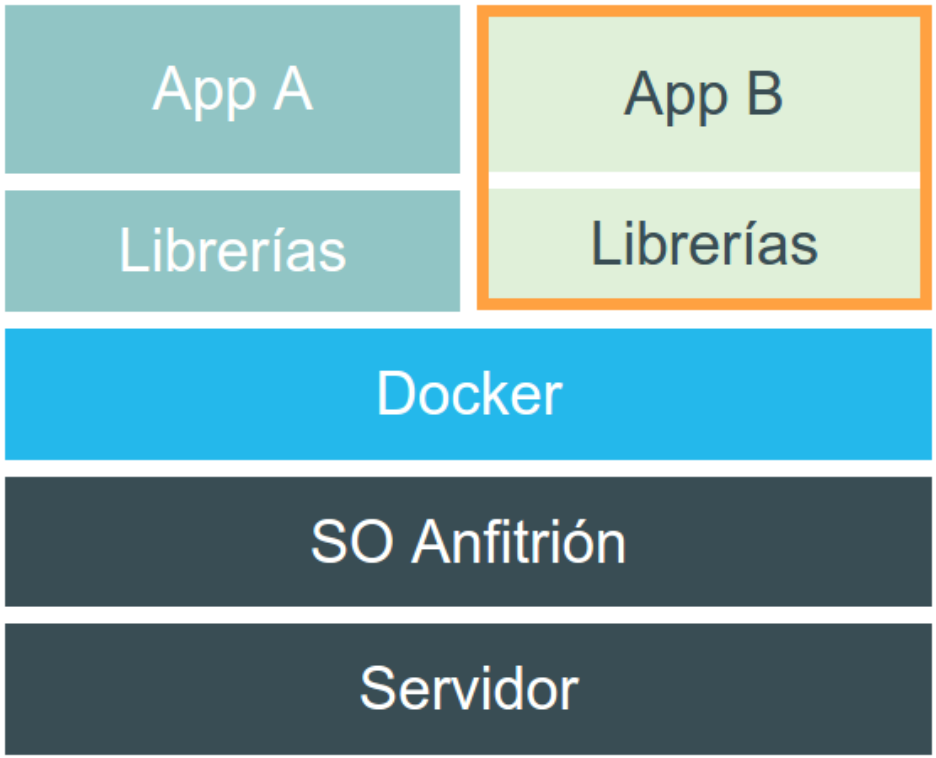
- Sistemas 64 bits
- Linux kernel 3.8+
- GoLang
- Linux container (LXC)

¿Cómo funciona? (frente a VMs)

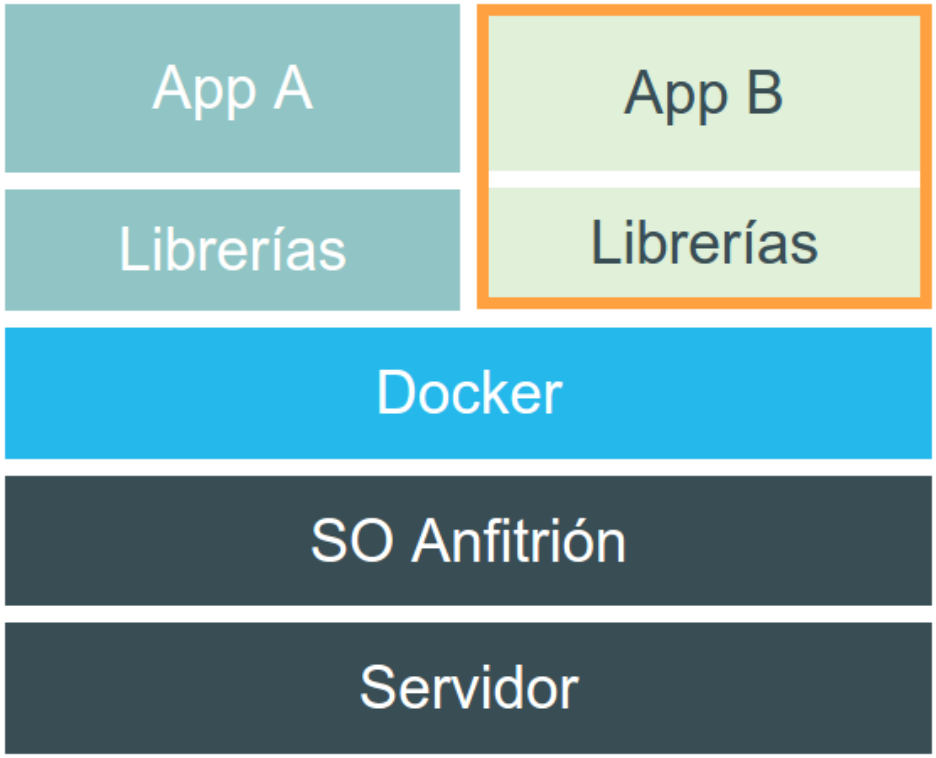
- Máquinas virtuales
  - SO Anfitrión
  - Hipervisor
  - SO Invitado
  - Librerías
  - Aplicación

¿Cómo funciona? (frente a VMs)

- Docker
  - SO Anfitrión
  - Docker
  - Librerías
  - Aplicación



¿Cómo funciona? (frente a VMs)



¿Por qué para desarrolladores?

- Sin problemas de dependencias.
- Cualquier lenguaje de programación.
- Aplicaciones portables multiplataforma.
- Control de versiones.
- Repositorios públicos o privados.
- Compilaciones automatizadas.

¿Por qué para administradores?

- Abstracción del SO anfitrión.
- Escalabilidad rápida.
- Seguridad añadida.
- Máximo rendimiento.
- Menor tamaño de los contenedores.
- Mayor aprovechamiento del hardware.
- Funciona dentro de máquinas virtuales.

Ventajas

- Docker es código abierto
  - https://github.com/docker/docker
- Sistema enfocado a aplicaciones, no a máquinas.
- Reutilización de compilaciones previas.
- Ecosistema en crecimiento (Docker Hub, StackOverFlow…).
- Aislamiento de recursos (sistema de archivos, red, ram…).

¡Aún más!

- Linux container daemon (LXD)
  - Hecho por Ubuntu/Canonical.
  - Hipervisor para contenedores.
  - Migración en caliente entre distintas máquinas.
  - Seguridad a nivel de hardware.
  - Gestión avanzada de redes.
  - API REST limpia y extensible.

¡Aún más!

- Kubernetes
  - Hecho por Google.
  - Gestor de clústeres de contenedores.
  - API de bajo nivel.
  - Mecanismos de self-healing:
    - Reinicios automáticos.
    - Programación de eventos.
    - Clonación/réplicas.

Introducción a Docker - Parte 2

@javierprovecho

Demostraciones

- Ubuntu 14.10
- LAMP stack
- Wordpress
- Cloud 9 IDE
- Fedora Desktop

```
docker run -t -i ubuntu:utopic
```


```
docker run -p 80:80 -p 3306:3306 tutum/lamp
curl http://localhost/
```

```
docker run -d -p 80:80 tutum/wordpress
curl http://localhost/
```

```
docker run -d -p 3131:3131 -v /tmp:/workspace alanct/cloud9
```

```
docker run -d -p 3389:3389 jumanjiman/xrdp
foo:bar
```

Introducción a Docker - Parte 3

@javierprovecho

Paso 1.1 - Cómo instalar Docker

```
# curl -sSL https://get.docker.com/ubuntu/ | sh
Get:29 http://security.ubuntu.com trusty-security/multiverse Sources [698 B]
Get:30 http://security.ubuntu.com trusty-security/main i386 Packages [147 kB]
Get:31 http://security.ubuntu.com trusty-security/restricted i386 Packages [14 B]
Get:32 http://security.ubuntu.com trusty-security/universe i386 Packages [51.2 kB]
Get:33 http://security.ubuntu.com trusty-security/multiverse i386 Packages [1,401 B]
Get:34 http://security.ubuntu.com trusty-security/main Translation-en [75.0 kB]
Hit http://security.ubuntu.com trusty-security/multiverse Translation-en
Hit http://security.ubuntu.com trusty-security/restricted Translation-en
Get:35 http://security.ubuntu.com trusty-security/universe Translation-en [30.5 kB]
Fetched 1,700 kB in 5s (311 kB/s)
Reading package lists... Done
...
```

Paso 1.2 - Cómo acceder a Docker

```
# docker
Usage: Docker [OPTIONS] COMMAND [arg...]
-H="127.0.0.1:4243": Host:port to bind/connect to

A self-sufficient runtime for linux containers.

Commands:

attach    Attach to a running container
build     Build a container from a Dockerfile
commit    Create a new image from a container's changes
diff      Inspect changes on a container's filesystem
export    Stream the contents of a container as a tar archive
history   Show the history of an image
images    List images
import    Create a new filesystem image from the contents of a tarball
info      Display system-wide information
insert    Insert a file in an image
inspect   Return low-level information on a container
...
```

Paso 2.1 - Buscar imágenes

```
# docker search tutorial
Found 1 results matching your query ("tutorial")
NAME                      DESCRIPTION
learn/tutorial            An image for the interactive tutorial
```

Paso 2.2 - Descargar imágenes

```
# docker pull learn/tutorial
Pulling repository learn/tutorial from https://index.docker.io/v1
Pulling image 8dbd9e392a964056420e5d58ca5cc376ef18e2de93b5cc90e868a1bbc8318c1c (precise)
from ubuntu
Pulling image b750fe79269d2ec9a3c593ef05b4332b1d1a02a62b4accb2c21d589ff2f5f2dc (12.10) from
ubuntu
Pulling image 27cf784147099545 () from tutorial
```

Paso 3 - Ejecutar comandos

```
# docker run learn/tutorial echo “Hola mundo”
Hola mundo
```

Paso 4 - Instalar paquetes

```
# docker run learn/tutorial apt-get install -y ping
Reading package lists...
Building dependency tree...
The following NEW packages will be installed:
  iputils-ping
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 56.1 kB of archives.
After this operation, 143 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu/ precise/main iputils-ping amd64 3:20101006-1ubuntu1
[56.1 kB]
debconf: delaying package configuration, since apt-utils is not installed
Fetched 56.1 kB in 1s (50.3 kB/s)
Selecting previously unselected package iputils-ping.
(Reading database ... 7545 files and directories currently installed.)
Unpacking iputils-ping (from .../iputils-ping_3%3a20101006-1ubuntu1_amd64.deb) ...
Setting up iputils-ping (3:20101006-1ubuntu1) ...
```

Paso 4 - Instalar paquetes

```
# docker run learn/tutorial apt-get install -y ping
Reading package lists...
Building dependency tree...
The following NEW packages will be installed:
  iputils-ping
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 56.1 kB of archives.
After this operation, 143 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu/ precise/main iputils-ping amd64 3:20101006-1ubuntu1
[56.1 kB]
debconf: delaying package configuration, since apt-utils is not installed
Fetched 56.1 kB in 1s (50.3 kB/s)
Selecting previously unselected package iputils-ping.
(Reading database ... 7545 files and directories currently installed.)
Unpacking iputils-ping (from .../iputils-ping_3%3a20101006-1ubuntu1_amd64.deb) ...
Setting up iputils-ping (3:20101006-1ubuntu1) ...
```

Paso 5 - Guardar cambios

```
# docker ps
ID                  IMAGE               COMMAND             CREATED             STATUS

# docker ps -a
ID                  IMAGE               COMMAND              CREATED             STATUS
6982a9948422        ubuntu:12.04        apt-get install ping   1 minute ago        Exit 0

# docker commit 698 learn/ping
effb66b31edb
```

Paso 6 - Ejecutar la nueva imagen

```
# docker run learn/ping ping inf.uva.es
PING inf.uva.es (157.88.109.216) 56(84) bytes of data.
64 bytes from www.inf.uva.es (157.88.109.216): icmp_seq=1 ttl=55 time=55.1 ms
64 bytes from www.inf.uva.es (157.88.109.216): icmp_seq=2 ttl=55 time=53.7 ms
64 bytes from www.inf.uva.es (157.88.109.216): icmp_seq=3 ttl=55 time=55.4 ms
64 bytes from www.inf.uva.es (157.88.109.216): icmp_seq=4 ttl=55 time=54.6 ms
64 bytes from www.inf.uva.es (157.88.109.216): icmp_seq=5 ttl=55 time=55.1 ms
64 bytes from www.inf.uva.es (157.88.109.216): icmp_seq=6 ttl=55 time=55.6 ms
64 bytes from www.inf.uva.es (157.88.109.216): icmp_seq=7 ttl=55 time=53.6 ms
64 bytes from www.inf.uva.es (157.88.109.216): icmp_seq=8 ttl=55 time=56.2 ms
64 bytes from www.inf.uva.es (157.88.109.216): icmp_seq=9 ttl=55 time=53.2 ms
...
```

Paso 7 - Inspeccionar el contenedor

```
# docker inspect efe
[2014/11/06 00:00:01 GET /v1.3/containers/efef/json
{
  "ID": "efefdc74a1d5900d7d7a74740e5261c09f5f42b6dae58ded6a1fde1cde7f4ac5",
  "Created": "2014-11-06T00:00:01.417119736Z",
  "Path": "ping",
  "Args": [
      "inf.uva.es"
  ],
  "Config": {
      "Hostname": "efefdc74a1d5",
      "User": "",
      "Memory": 0,
      "MemorySwap": 0,
      "CpuShares": 0,
      "AttachStdin": false,
      "AttachStdout": true,
      "AttachStderr": true,
      "PortSpecs": null,
      "Tty": false,
      "OpenStdin": false,
      "StdinOnce": false,
      "Env": null,
...
```

Paso 8 - Dockerfile

```
# nano Dockerfile
FROM ubuntu:latest
RUN apt-get -y update
RUN apt-get -y upgrade
RUN apt-get -y install git python
RUN echo “Hola mundo” >> index.html
EXPOSE 80
CMD [“python”, “-m”, “SimpleHTTPServer”, “80”]

# docker build -t learn/webserver .
...

# docker run -P learn/webserver
Serving HTTP on 0.0.0.0 port 80 ...
```

Paso 9.1 - Guardar/cargar imágenes

```
# docker save learn/ping > /tmp/imagen.tar
...

# docker load < /tmp/imagen.tar
...
```

Paso 9.2 - Exportar/importar conten.

```
# docker ps -a
ID                  IMAGE               COMMAND                 CREATED             STATUS
6982a9948422        ubuntu:12.04        apt-get install ping    1 minute ago        Exit 0

# docker export 698 > /tmp/contenedor.tar
...

# cat /tmp/contenedor.tar | docker import - learn/ping-export:latest
...
```

Introducción a Docker - Parte 4

@javierprovecho

Enlaces

Crédito $10 en servicios de la nube.

http://goo.gl/yaXF8Y

Licencia

Agradecimientos

Escuela Técnica Superior de Ingeniería Informática de Valladolid

Grupo Universitario de Informática de Valladolid