

- * Instalar Docker Toolbox
 - * Buscar y Descargar Imagen
 - * Crear contenedor a partir de imagen
 - * Conectar por IP y puerto elegido
-

* EN CUALQUIER CASO HAY UNOS PRERREQUISITOS QUE CUMPLIR:

- * 64 bits
- * 4gb ram
- * Al menos 20Gb disco duro

DISTRIBUCIONES LINUX

- * En linux es muy sencillo no necesitamos docker toolbox en el siguiente enlace encontramos las instrucciones para instalarlo en distribuciones linux

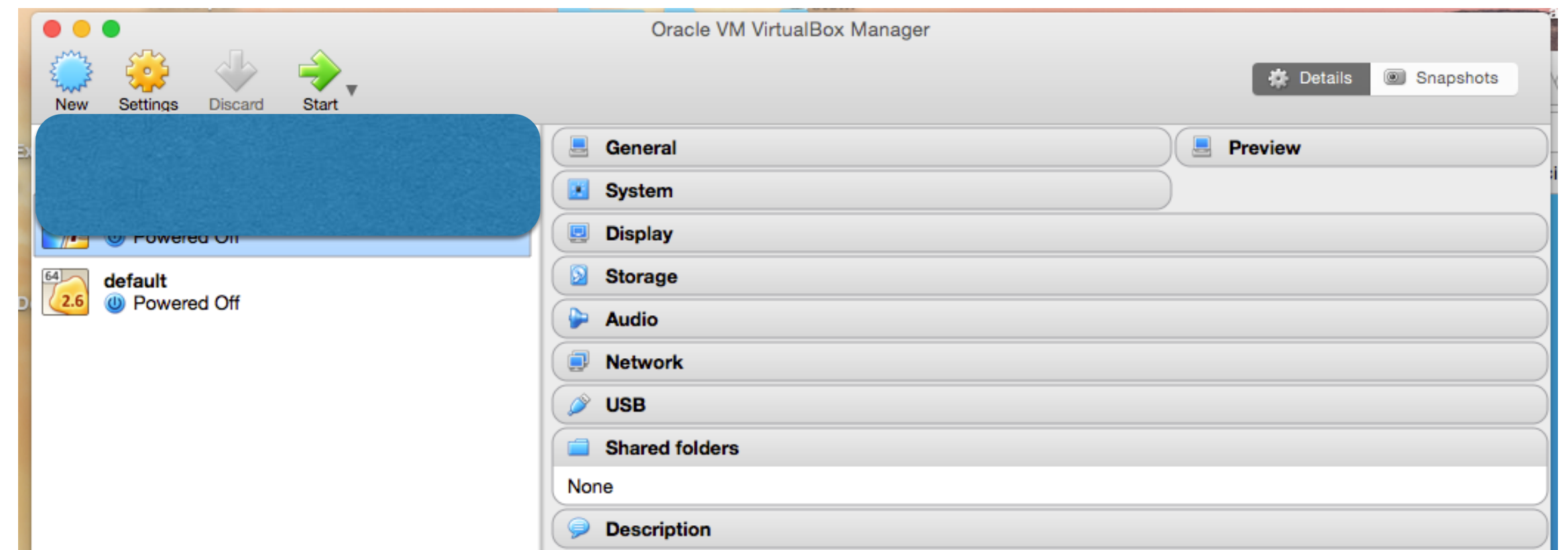
<https://docs.docker.com/engine/installation/linux/>

Install docker toolbox

WINDOWS

MAC

- * Docker trabaja sobre linux es necesario utilizar máquina virtual de linux.
- * La toolbox de Docker ya nos proporciona un framework para realizar estos pasos de forma transparente
- * Por lo tanto tenemos dos opciones:
 - Montar una imagen de linux en cualquier máquina virtual e instalar como anteriormente se ha comentado (Si la monta kitematic tendrá el nombre default)
 - Utilizar kitematic, este framework que nos facilita la tarea <https://kitematic.com/>



LINUX

- * Para descargar imagen del repositorio global utilizamos el comando pull

```
$ docker pull debian

Using default tag: latest
latest: Pulling from library/debian
fdd5d7827f33: Pull complete
a3ed95caeb02: Pull complete
Digest: sha256:e7d38b3517548a1c71e41bffe9c8ae6d6d29546ce46bf62159837aad072c90aa
Status: Downloaded newer image for debian:latest
```

- * Con images puedes ver las que están descargadas

```
$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
debian	jessie	f50f9524513f	5 days ago	125.1 MB
debian	latest	f50f9524513f	5 days ago	125.1 MB

EN LINUX LA IP DE CONEXIÓN CON DOCKER SUELE SER:

172.17.0.1

BUSCAR Y DESCARGAR IMAGEN

WINDOWS

MAC

- ✧ Podemos descargar la imagen por dos medios mediante linea de comando en linux como se ha explicado o utilizando kitematic como explicaremos a continuación.

Máquina virtual linux con docker:

[illegible]

- * En este caso abrimos la ventana de comandos de la máquina virtual.
- * Y trabajamos sobre Docker
- * OBSERVAR QUE NOS INDICA CÚAL ES LA IP QUE SE HA ASIGNADO

WINDOWS

MAC

* Si no nos hemos fijado en que nos indica la ip podemos buscarla utilizando ifconfig

Máquina virtual linux con docker:

```
default [Running]
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:70 errors:0 dropped:0 overruns:0 frame:0
TX packets:79 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:7650 (7.4 KiB)  TX bytes:7269 (7.0 KiB)

eth1  Link encap:Ethernet  HWaddr 08:00:27:EC:BC:35
      inet addr:192.168.99.100  Bcast:192.168.99.255  Mask:255.255.255.0
      inet6 addr: fe80::a00:27ff:feec:bc35/64 Scope:Link
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:7 errors:0 dropped:0 overruns:0 frame:0
TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:1610 (1.5 KiB)  TX bytes:1333 (1.3 KiB)

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:32 errors:0 dropped:0 overruns:0 frame:0
TX packets:32 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1
RX bytes:2752 (2.6 KiB)  TX bytes:2752 (2.6 KiB)

root@default:~#
```

* Suele tener la forma 192.168.99.100

WINDOWS

MAC

* Una vez conectados introducimos los comandos descritos para linux, en nuestro caso:

Máquina virtual linux con docker:

```
root@default:~# docker pull gvalverd/dl
Using default tag: latest
latest: Pulling from gvalverd/dl
Digest: sha256:a2c898852c04cba5ad4f313343d7de0ec4a054bdf55be391a8e20e56713fd766
Status: Image is up to date for gvalverd/dl:latest
root@default:~# _
```

* Observar que tenemos:

- Dirección del repositorio en hub de docker: gvalverd/dl (usuario/imagen)
- Nombre de la versión del mismo: tag latest (version tomada por defecto)
 - Si quisieramos descargar otra: **docker pull gvalverd/dl:newTag**

Crear contenedor a partir de imagen

WINDOWS

MAC

* Una vez conectados introducimos los comandos descritos para linux, en nuestro caso:

Máquina virtual linux con docker:

```
bash-3.2$ docker run -i -p 8888:54111 gvalverd/dl
[I 11:58:19.765 NotebookApp] Writing notebook server cookie secret to /root/.local/share/jupyter/runtime/notebook_cookie_secret
/usr/local/lib/python2.7/dist-packages/widgetsnbextension/__init__.py:30: UserWarning: To use the jupyter-js-widgets nbextension, you'll need to update
the Jupyter notebook to version 4.2 or later.
the Jupyter notebook to version 4.2 or later.""")
[W 11:58:19.829 NotebookApp] WARNING: The notebook server is listening on all IP addresses and not using encryption. This is not recommended.
[W 11:58:19.830 NotebookApp] WARNING: The notebook server is listening on all IP addresses and not using authentication. This is highly insecure and not recommended.
[I 11:58:19.837 NotebookApp] Serving notebooks from local directory: /notebooks
[I 11:58:19.837 NotebookApp] 0 active kernels
[I 11:58:19.838 NotebookApp] The Jupyter Notebook is running at: http://[all ip addresses on your system]:8888/
[I 11:58:19.838 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
```

* Observar que tenemos:

- Podemos ver que los mensajes nos informa de como conectarnos al jupyter notebook para trabajar
- El parámetro -p nos permite asignar el puerto interno 8888 al puerto 54111
 - **Debemos comprobar que 54111 está abierto en nuestro sistema**

BUSCAR Y DESCARGAR IMAGEN

WINDOWS

MAC

Kitematic:

BUSCAMOS EL REPOSITORIO
gvalverd

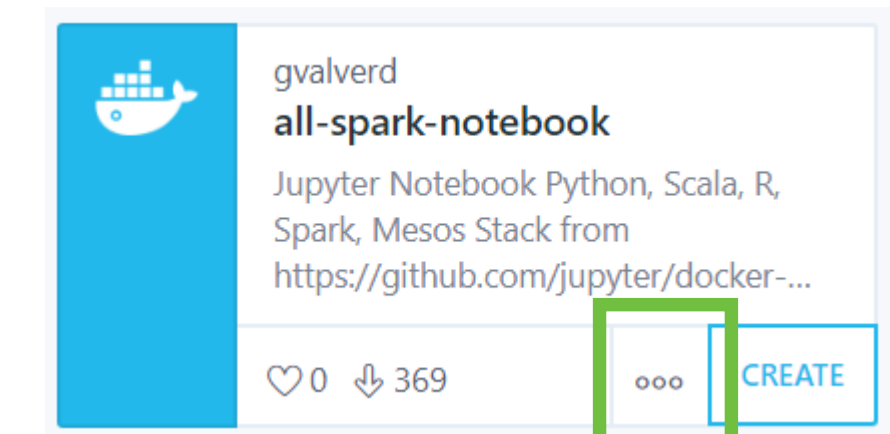


Nos abre una ventana de comandos con Docker

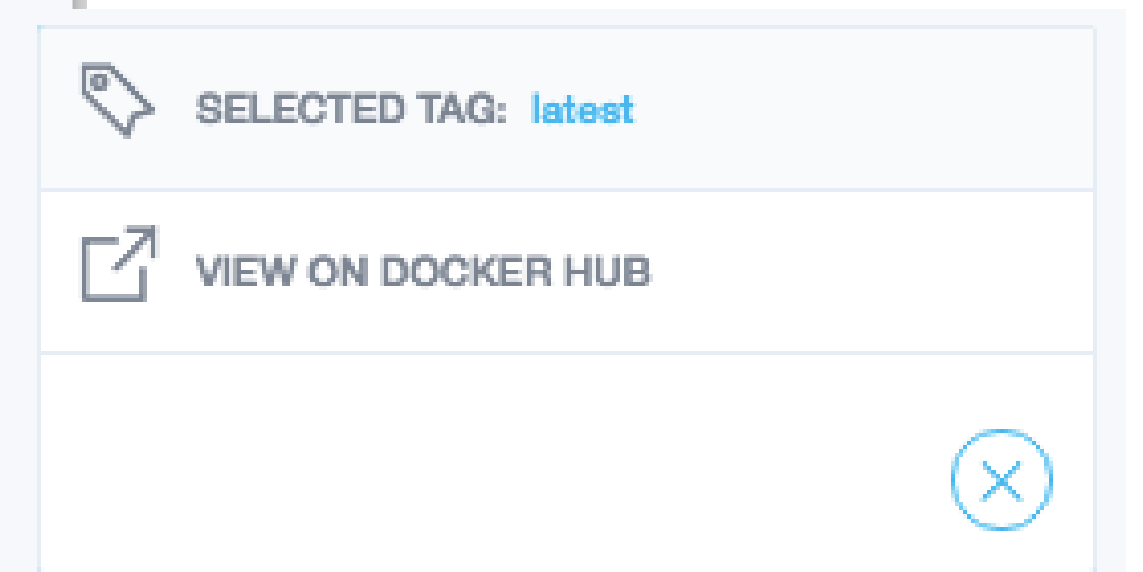
Al crear el contenedor desde esta imagen automaticamente:
SE DESCARGA LA VERSION LATEST
INSTALA EL CONTENEDOR

OJO: Si al pulsar create no empieza a descargar o se atasca, CLICAMOS AQUÍ

Se abrirá una consola y lanzaremos los comandos mencionados previamente



Podemos seleccionar que tag queremos y ir a la web del repositorio

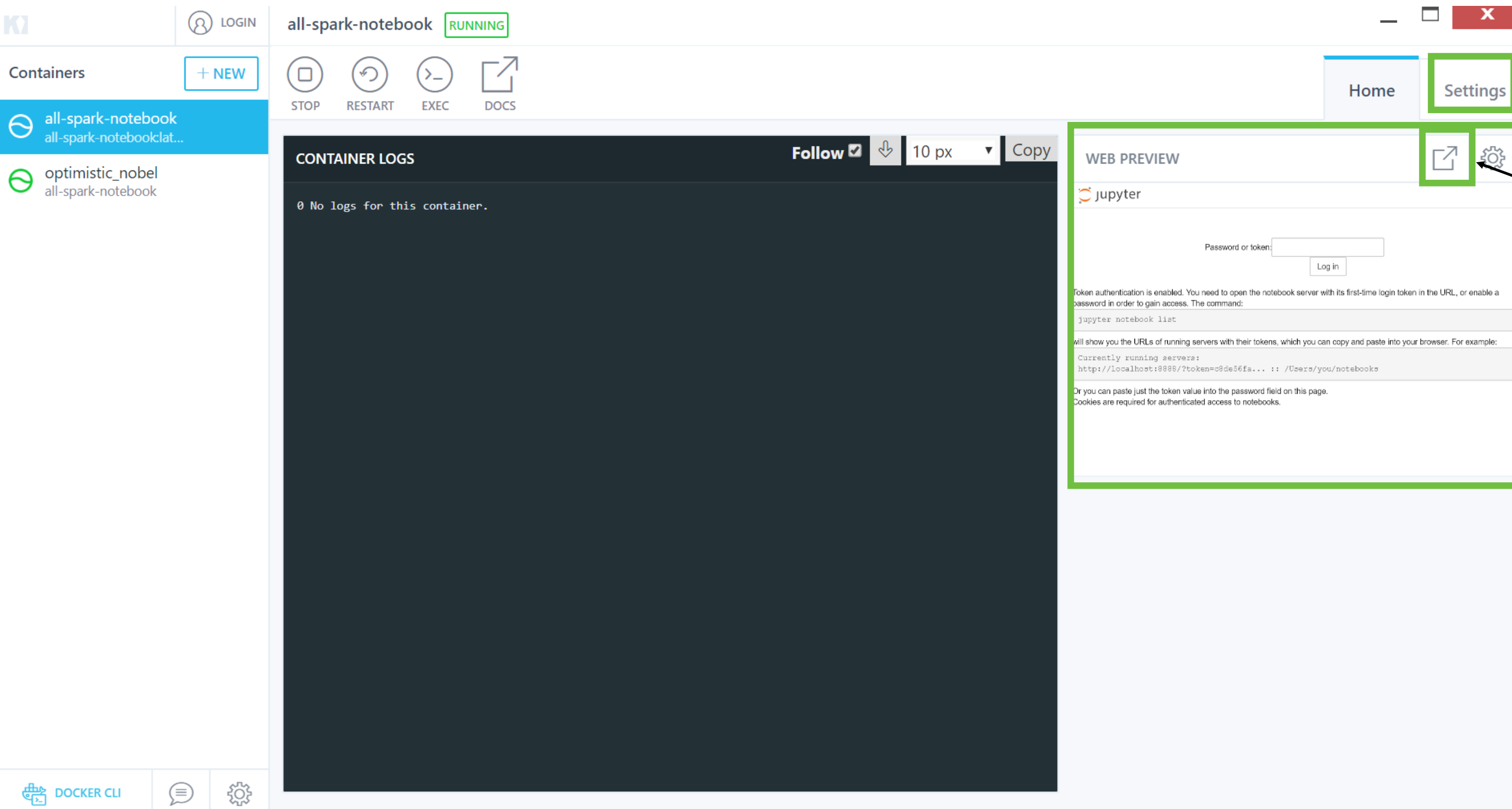


Crear contenedor a partir de imagen

WINDOWS

MAC

Kitematic:



Podemos ver que nos aparece una ventana con la vista previa de la ventana de jupyter.

Pinchando en este botón nos manda al navegador donde accedemos

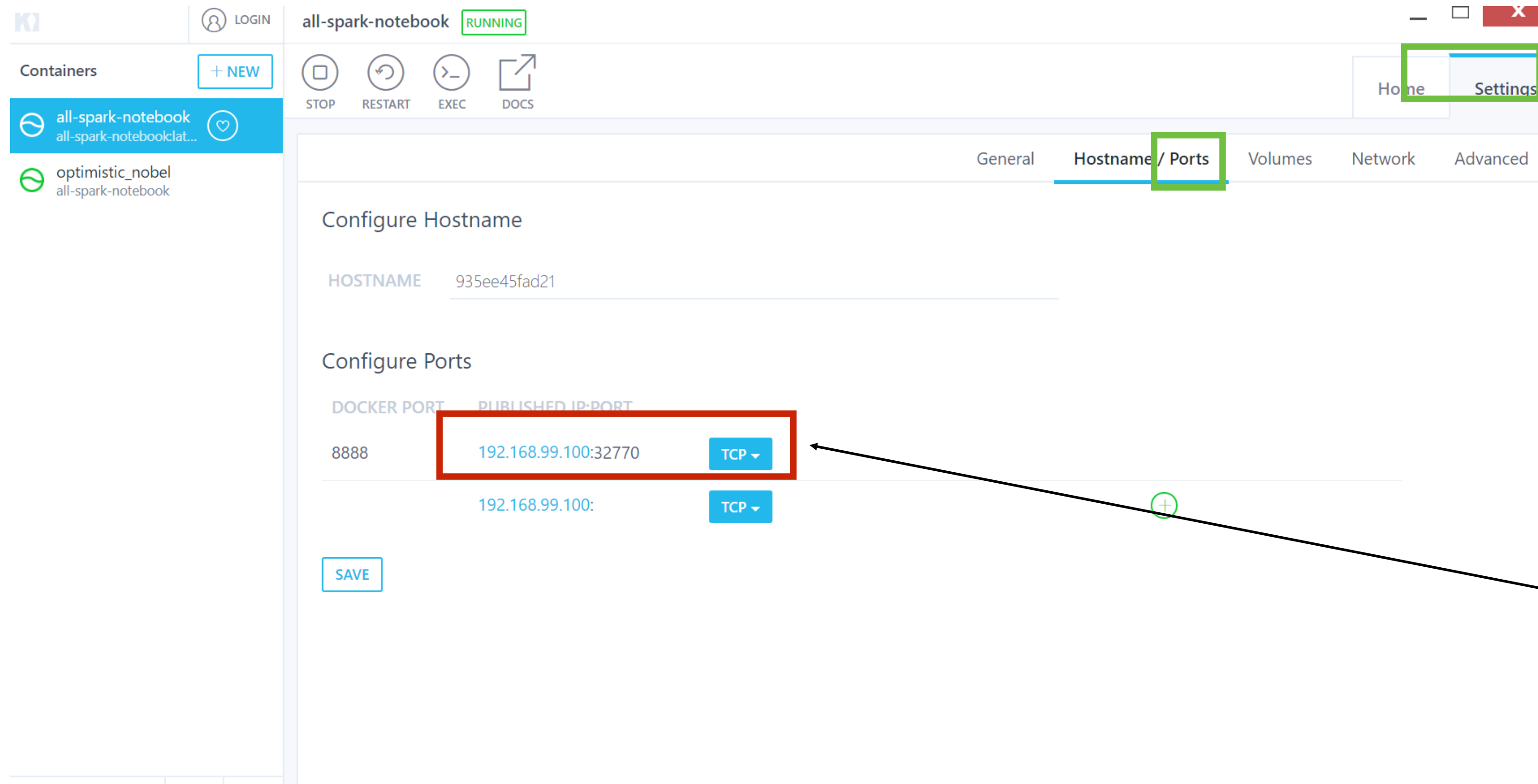
Si no apareciera puede ser por una mala configuración de los puertos en **SETTINGS** podemos configurarlos

Crear contenedor a partir de imagen

WINDOWS

MAC

Kitematic:



Dónde está el token

WINDOWS

MAC

LINUX

* Deberíamos poder llegar a esto con la ip y el puerto correspondiente:

The screenshot shows a JupyterLab interface. On the left, a terminal window displays the output of the command `jupyter notebook list`. The output shows a running server with a token: `http://localhost:8888/?token=c8de56fa... :: /Users/you/notebooks`. On the right, a container management panel shows a list of containers. One container named `all-spark-notebook` is highlighted. An arrow points from the `EXEC` button in the container management panel to the terminal window.

192.168.99.100:32770/login

jupyter

Password or token Log in

Token authentication is enabled. You need to open the notebook server with its first-time login token in the URL, or enable a password in order to gain access. The command:

```
jupyter notebook list
```

will show you the URLs of running servers with their tokens, which you can copy and paste into your browser. For example:

```
Currently running servers:
http://localhost:8888/?token=c8de56fa... :: /Users/you/notebooks
```

Or you can paste just the token value into the password field on this page. Cookies are required for authenticated access to notebooks.

Containers

+ NEW

all-spark-notebook all-spark-notebookclat...

optimistic_nobel all-spark-notebook

STOP RESTART EXEC DOCS

CONTAINER LOGS

Follow 10 px Copy

0 No logs for this container.

WEB PREVIEW

jupyter

Password or token: Log in

Token authentication is enabled. You need to open the notebook server with its first-time login token in the URL, or enable a password in order to gain access. The command:

```
jupyter notebook list
```

will show you the URLs of running servers with their tokens, which you can copy and paste into your browser. For example:

```
Currently running servers:
http://localhost:8888/?token=c8de56fa... :: /Users/you/notebooks
```

Or you can paste just the token value into the password field on this page. Cookies are required for authenticated access to notebooks.

```
jovyan@935ee45fad21: ~/work
jovyan@935ee45fad21:~/work$ jupyter notebook list
Currently running servers:
http://localhost:8888/?token=73019034f621f708c177854912eab482f906835228400fa7 : /home/jovyan/work
jovyan@935ee45fad21:~/work$
```

DOCKER CLI

Si no conocemos el token:

Debería aparecer en kitematic

Si no:

1 Pulsamos exec

2 Introducimos en la terminal:

jupyter notebook list

3 Introducir el código de la sesión que corresponda

Conectar por IP y puerto elegido

WINDOWS

MAC

LINUX

* Deberíamos poder llegar a esto con la ip y el puerto correspondiente:

