

netstat

VIA: `man netstat`

Print network connections, routing tables, interface statistics, masquerade connections, and multicast memberships

`-a, --all`

Show both listening and non-listening sockets. With the `--interfaces` option, show interfaces that are not up

`-n --numeric`

Show numerical addresses instead of trying to determine symbolic host, port or user names.

useful information about unconfigured address families.

`-o --timers`

Include information related to networking timers.

`-p, --program`

Show the PID and name of the program to which each socket belongs.

USO: `netstat [options]`

Permite mostrar diferentes elementos de la red tales como conexiones, tablas de enrutado, estadísticas de interfaz, máscara de conexiones y miembros de multicast.

En nuestro caso de ejecución emplearemos las opciones `"-vap"` que serán modo **verbose** (con más información), los **sockets** que están y no a la **escucha** y finalmente el **PID** del proceso.

docker run

VIA: <https://docs.docker.com/engine/reference/commandline/run/>

`-d, --detach` Run container in background and print container ID

`-p, --publish` Publish a container's port(s) to the host

`--name` Assign a name to the container

USO: `docker run [OPTIONS] IMAGE [COMMAND] [ARG...]`

Este comando ya lo hemos visto anteriormente, nos permite crear un contenedor

y lanzarlo. Las opciones que va a emplear nuestro comando será *-d*, *-p* y *--name*. La primera permite lanzar un contenedor en **segundo plano**, la segunda opción permite **publicar el puerto** del contenedor para que podamos acceder desde fuera del mismo y la tercera opción nos permite darle un **nombre concreto** al contenedor.