EJERCICIOS TEMA 1 DOCKER

> *Realizado por Pablo R.*

URL de Fuentes

https://docs.docker.com/engine/install/ubuntu/

1. **Instala Docker en una máquina y configúralo para que se pueda usar con un usuario sin privilegios.**

   Inicializamos el repositorio de Docker tal y como se nos indica en la la web oficial.

```
apt-get install ca-certificates curl gnupg lsb-release
```

```
sudo mkdir -p /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
```

```
echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
root@pablo-vm:/home/pablo# apt install ca-certificates curl gnupg lsb-release
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
lsb-release is already the newest version (11.1.0ubuntu4).
lsb-release set to manually installed.
ca-certificates is already the newest version (20211016ubuntu0.22.04.1).
ca-certificates set to manually installed.
gnupg is already the newest version (2.2.27-3ubuntu2.1).
gnupg set to manually installed.
The following packages were automatically installed and are no longer required:
  libflashrom1 libftdi1-2
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  curl
0 upgraded, 1 newly installed, 0 to remove and 1 not upgraded.
Need to get 193 kB of archives.
After this operation, 453 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://es.archive.ubuntu.com/ubuntu jammy-updates/main amd64 curl amd64 7.81.0-1ubuntu1.7 [193 kB]
Fetched 193 kB in 6s (30,0 kB/s)
Selecting previously unselected package curl.
(Reading database ... 202275 files and directories currently installed.)
Preparing to unpack .../curl_7.81.0-1ubuntu1.7_amd64.deb ...
Unpacking curl (7.81.0-1ubuntu1.7) ...
Setting up curl (7.81.0-1ubuntu1.7) ...
Processing triggers for man-db (2.10.2-1) ...
root@pablo-vm:/home/pablo# sudo mkdir -p /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
root@pablo-vm:/home/pablo# echo \
  "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/ubuntu \
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
root@pablo-vm:/home/pablo#
```

Realizamos la instalacion

```
sudo apt update
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

```
root@pablo-vm:/home/pablo# apt update
Hit:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:3 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease
Get:4 https://download.docker.com/linux/ubuntu jammy InRelease [48,9 kB]
Get:5 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [11,2 kB]
Hit:6 http://security.ubuntu.com/ubuntu jammy-security InRelease
Fetched 60,1 kB in 7s (8.879 B/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
1 package can be upgraded. Run 'apt list --upgradable' to see it.
root@pablo-vm:/home/pablo# sudo apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libflashrom1 libftdi1-2
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  docker-ce-rootless-extras docker-scan-plugin git git-man liberror-perl libslirp0 pigz slirp4netns
Suggested packages:
  aufs-tools cgroupfs-mount | cgroup-lite git-daemon-run | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  containerd.io docker-ce docker-ce-cli docker-ce-rootless-extras docker-compose-plugin docker-scan-plugin git git-man liberror-perl libslirp0 pigz
  slirp4netns
0 upgraded, 12 newly installed, 0 to remove and 1 not upgraded.
Need to get 116 MB of archives.
After this operation, 449 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

2. **Ejecuta un contenedor a partir de la imagen hello-word. Comprueba que nos devuelve la salida adecuada. Comprueba que no se está ejecutando. Lista los contenedores que están parado. Borra el contenedor.**

```
docker pull hello-world
docker run hello-world
docker ps -a
```

```
root@pablo-vm:/home/pablo# docker pull hello-world
Using default tag: latest
latest: Pulling from library/hello-world
2db29710123e: Pull complete
Digest: sha256:aa0cc8055b82dc2509bed2e19b275c8f463506616377219d9642221ab53cf9fe
Status: Downloaded newer image for hello-world:latest
docker.io/library/hello-world:latest
root@pablo-vm:/home/pablo# docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/
```

Descargamos y ejecutamos el contenedor hello-world y observamos que nos devuelve la salida correctamente.

```
docker ps -a
docker rm <id_contenedor>
```

```
root@pablo-vm:/home/pablo# docker ps -a
CONTAINER ID   IMAGE         COMMAND    CREATED         STATUS                     PORTS     NAMES
81911e108e07   hello-world   "/hello"   4 minutes ago   Exited (0) 4 minutes ago             inspiring_volhard
root@pablo-vm:/home/pablo# docker rm 81911e108e07
81911e108e07
root@pablo-vm:/home/pablo# docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES
root@pablo-vm:/home/pablo#
```

Observamos que el contenedor se cierra una vez termina su ejecución Exited (0). Procedemos a borrarlo y comprobamos que este borrado correctamente.

3. **Crea un contenedor interactivo desde una imagen debian. Instala un paquete (por ejemplonano). Sal de la terminal, ¿sigue el contenedor corriendo? ¿Por qué?. Vuelve a iniciar elcontenedor y accede de nuevo a él de forma interactiva. ¿Sigue instalado el nano?. Sal delcontenedor, y bórralo. Crea un nuevo contenedor interactivo desde la misma imagen. ¿Tiene el nano instalado?**

```
docker pull debian
```

```
root@pablo-vm:/home/pablo# docker pull debian
Using default tag: latest
latest: Pulling from library/debian
bbeef03cda1f: Pull complete
Digest: sha256:534da5794e770279c889daa891f46f5a530b0c5de8bfbc5e40394a0164d9fa87
Status: Downloaded newer image for debian:latest
docker.io/library/debian:latest
root@pablo-vm:/home/pablo#
```

Obtenemos la imagen de debian con pull, para posteriormente poder ejecutar el sistema dockerizado.

```
docker run -it --name debian1 debian bash
apt update
apt install nano
exit
```

```
root@pablo-vm:/home/pablo# docker run -it --name debian1 debian bash
root@47b6f7facd33:/# apt update
Get:1 http://deb.debian.org/debian bullseye InRelease [116 kB]
Get:2 http://deb.debian.org/debian-security bullseye-security InRelease [48.4 kB]
Get:3 http://deb.debian.org/debian bullseye-updates InRelease [44.1 kB]
Get:4 http://deb.debian.org/debian bullseye/main amd64 Packages [8183 kB]
Get:5 http://deb.debian.org/debian-security bullseye-security/main amd64 Packages [214 kB]
Get:6 http://deb.debian.org/debian bullseye-updates/main amd64 Packages [14.6 kB]
Fetched 8620 kB in 1s (8415 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
root@47b6f7facd33:/# apt install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libgpm2 libncursesw6
Suggested packages:
  gpm hunspell
The following NEW packages will be installed:
  libgpm2 libncursesw6 nano
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 825 kB of archives.
After this operation, 3087 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
```

```
root@47b6f7facd33:/# apt install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libgpm2 libncursesw6
Suggested packages:
  gpm hunspell
The following NEW packages will be installed:
  libgpm2 libncursesw6 nano
0 upgraded, 3 newly installed, 0 to remove and 0 not upgraded.
Need to get 825 kB of archives.
After this operation, 3087 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://deb.debian.org/debian bullseye/main amd64 libncursesw6 amd64 6.2+20201114-2 [132 kB]
Get:2 http://deb.debian.org/debian bullseye/main amd64 nano amd64 5.4-2+deb11u2 [657 kB]
Get:3 http://deb.debian.org/debian bullseye/main amd64 libgpm2 amd64 1.20.7-8 [35.6 kB]
Fetched 825 kB in 0s (10.1 MB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package libncursesw6:amd64.
(Reading database ... 6661 files and directories currently installed.)
Preparing to unpack .../libncursesw6_6.2+20201114-2_amd64.deb ...
Unpacking libncursesw6:amd64 (6.2+20201114-2) ...
Selecting previously unselected package nano.
Preparing to unpack .../nano_5.4-2+deb11u2_amd64.deb ...
Unpacking nano (5.4-2+deb11u2) ...
Selecting previously unselected package libgpm2:amd64.
Preparing to unpack .../libgpm2_1.20.7-8_amd64.deb ...
Unpacking libgpm2:amd64 (1.20.7-8) ...
Setting up libgpm2:amd64 (1.20.7-8) ...
Setting up libncursesw6:amd64 (6.2+20201114-2) ...
Setting up nano (5.4-2+deb11u2) ...
update-alternatives: using /bin/nano to provide /usr/bin/editor (editor) in auto mode
update-alternatives: using /bin/nano to provide /usr/bin/pico (pico) in auto mode
Processing triggers for libc-bin (2.31-13+deb11u5) ...
root@47b6f7facd33:/# exit
exit
root@pablo-vm:/home/pablo#
```

Creamos el contenedor interactivo, con el comando de la parte superior. Realizamos la instalación de nano y salimos del contenedor.

```
docker ps
docker ps -a
```

```
root@pablo-vm:/home/pablo# docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED         STATUS                     PORTS     NAMES
47b6f7facd33   debian    "bash"    2 minutes ago   Exited (0) 31 seconds ago            debian1
root@pablo-vm:/home/pablo#
```

Verificamos que el contenedor no se continúa ejecutando, ya que era un sesión interactiva

```
docker start debian1
docker exec -it debian1 bash
apt install nano
```

```
root@pablo-vm:/home/pablo# docker exec -it debian1 bash
Error response from daemon: Container 47b6f7facd33b1eba3eae7beffeecf674b5ca8a55942d703ae5369f880bb763b is not running
root@pablo-vm:/home/pablo# docker start debian1
debian1
root@pablo-vm:/home/pablo# docker exec -it debian1 bash
root@47b6f7facd33:/#
```

```
root@47b6f7facd33:/# apt install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
nano is already the newest version (5.4-2+deb11u2).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@47b6f7facd33:/#
```

Puesto que el contenedor se encuentra apagado, lo ejecutamos con docker start y procedemos a levantar una sesion de bash. comprobamos que nano sigue instalado

4. **Crea un contenedor demonio con un servidor nginx, usando la imagen oficial de nginx. Al crear el contenedor, ¿has tenido que indicar algún comando para que lo ejecute? Accede al navegador web y comprueba que el servidor esta funcionando. Muestra los logs del contenedor.**

```
docker pull nginx
```

```
root@pablo-vm:/home/pablo# docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
8740c948ffd4: Pull complete
d2c0556a17c5: Pull complete
c8b9881f2c6a: Pull complete
693c3ffa8f43: Pull complete
8316c5e80e6d: Pull complete
b2fe3577faa4: Pull complete
Digest: sha256:b8f2383a95879e1ae064940d9a200f67a6c79e710ed82ac42263397367e7cc4e
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
root@pablo-vm:/home/pablo#
```

Nos descargamos la imagen de nginx en nuestro equipo.

```
docker run -it nginx bash
```

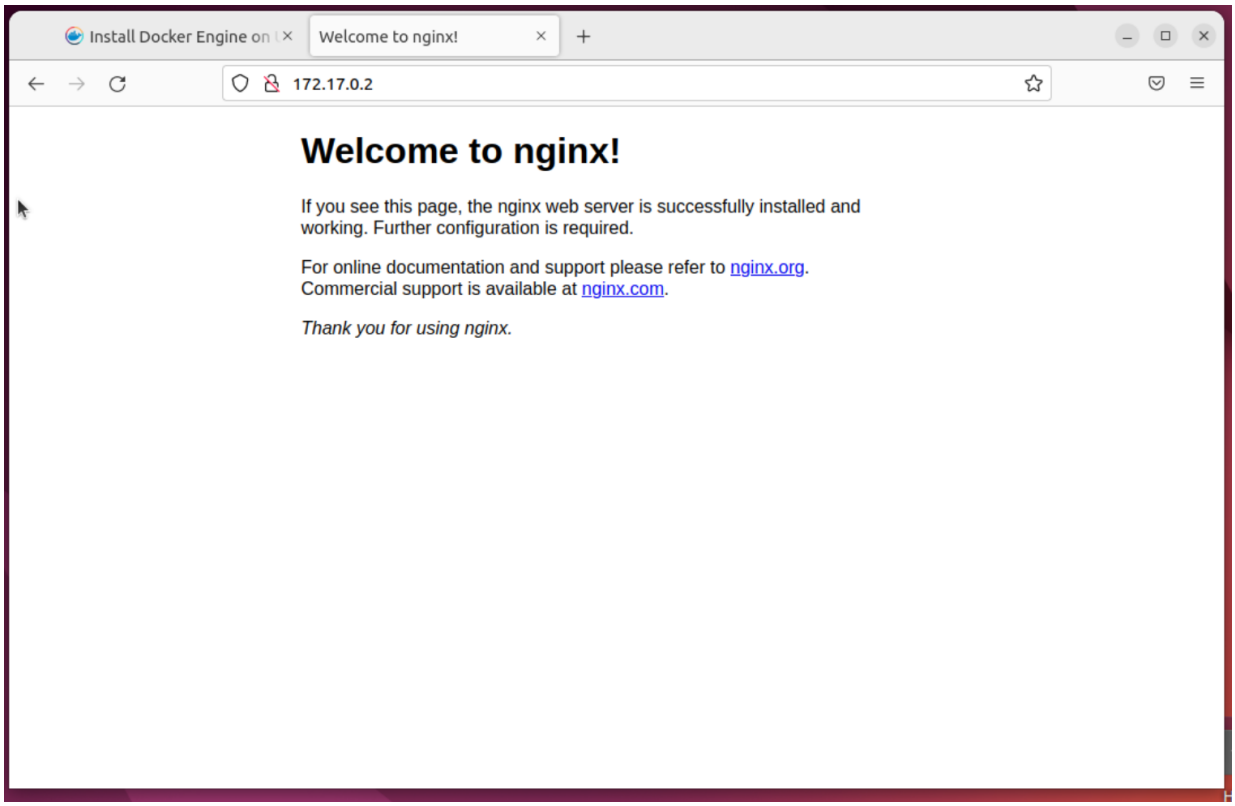
```
root@pablo-vm:/home/pablo# docker run -it nginx bash
root@b4580a984966:/#
```

Creamos el contenedor de nginx de forma interactiva.

```
docker inspect <codigoct>
```

```
            "GlobalIPv6PrefixLen": 0,
            "IPAddress": "172.17.0.2",
            "IPPrefixLen": 16,
            "IPv6Gateway": "",
            "MacAddress": "02:42:ac:11:00:02",
            "Networks": {
                "bridge": {
                    "IPAMConfig": null,
                    "Links": null,
                    "Aliases": null,
                    "NetworkID": "75abe74206c60f599729093a2d0bd1d01b4fa224ef9cd25b2c132cb128c75a4d",
                    "EndpointID": "d2f34485f20d569f175bce2d94690671cb667278c882a99062fc2925990f9dcd",
                    "Gateway": "172.17.0.1",
                    "IPAddress": "172.17.0.2",
                    "IPPrefixLen": 16,
                    "IPv6Gateway": "",
                    "GlobalIPv6Address": "",
                    "GlobalIPv6PrefixLen": 0,
                    "MacAddress": "02:42:ac:11:00:02",
                    "DriverOpts": null
                }
            }
        }
    }
]
root@pablo-vm:/home/pablo#
```


Utilizamos docker inpect para conocer la ip del contenedor y poder visualizar la web en el navegador.



Accedemos y visualizamos que la web nginx se muestra correctamente.

```
docker logs <idct>
```

```
root@pablo-vm:/home/pablo# docker logs a2a40af4d527
root@a2a40af4d527:/# nano /etc/nginx/nginx.conf
bash: nano: command not found
root@a2a40af4d527:/# apt install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
E: Unable to locate package nano
root@a2a40af4d527:/# apt update
```

Con *docker logs* vemos los últimos comandos ejecutados en el contenedor.

5. **Crea un contenedor con la aplicación Nextcloud, mirando la documentación en docker Hub,para personalizar el nombre de la base de datos sqlite que va a utilizar.**

```
docker pull nextcloud
```

```
root@pablo-vm:/home/pablo# docker pull nextcloud
Using default tag: latest
latest: Pulling from library/nextcloud
8740c948ffd4: Already exists
1873be858264: Pull complete
7ce6a163d8c1: Pull complete
008a172010ba: Pull complete
d15353ae3d77: Pull complete
223eb1888c0f: Pull complete
83374c2a967a: Pull complete
8fdc86711b26: Pull complete
23c0224c39b8: Pull complete
915d82c7f5c5: Pull complete
dc037a9c9035: Pull complete
768542e0b637: Pull complete
d7ade602d94f: Pull complete
7361225e9a5d: Pull complete
fdc75c5d6478: Pull complete
6e598d96642e: Pull complete
2183a95f6531: Pull complete
e4d461a63b9a: Pull complete
e50f69db5ce5: Pull complete
4e2f130d99f4: Pull complete
Digest: sha256:01435cc1463c01a8a611fd387dc588ccbd4abe9bf290d668c7d80eac160ab4ef
Status: Downloaded newer image for nextcloud:latest
docker.io/library/nextcloud:latest
root@pablo-vm:/home/pablo#
```

Obtenemos la imagen de netxcloud.

```
docker run -d -p 8080:80 -e SQLITE_DATABASE=minombre nextcloud
```

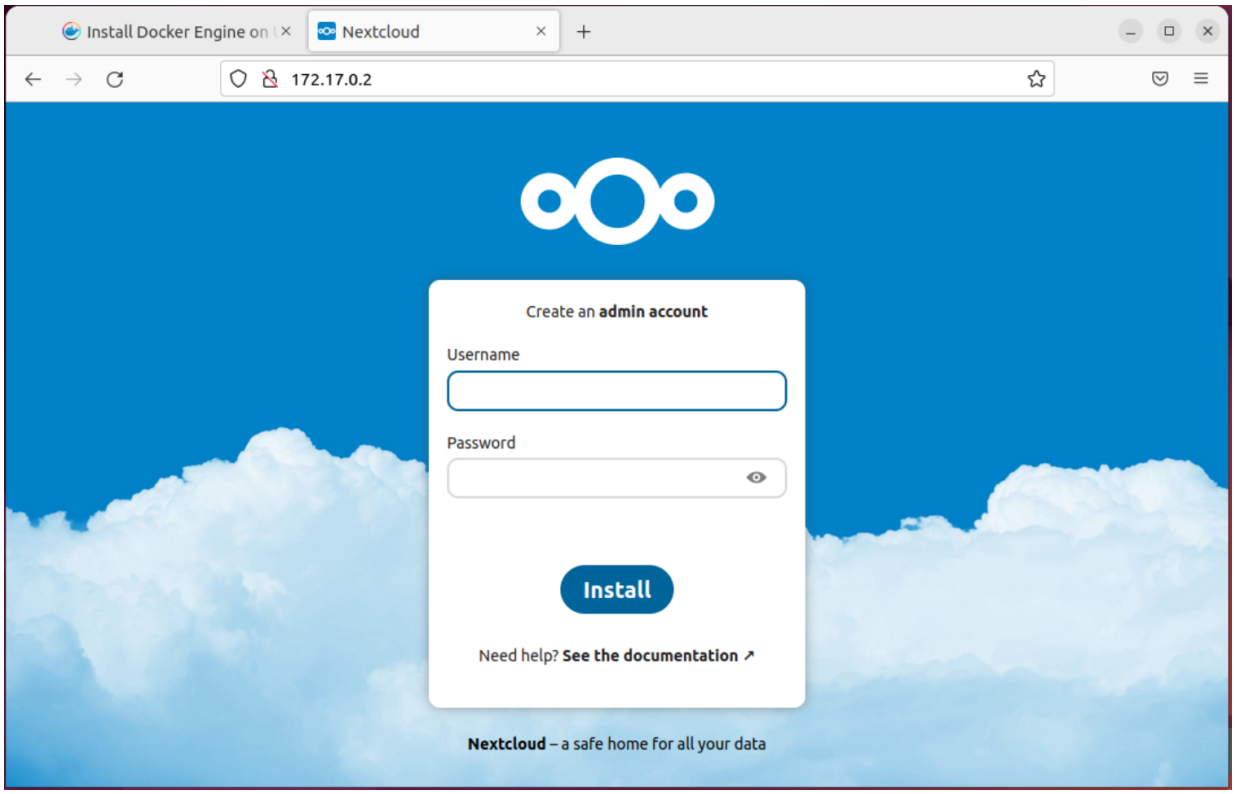
```
root@pablo-vm:/home/pablo# docker run -d -p 8080:80 -e SQLITE_DATABASE=minombre nextcloud
4b89dd306ce3970db73dc0c3cd908060c8dba359f7046f0e92cf1a3238137ae8
root@pablo-vm:/home/pablo#
```

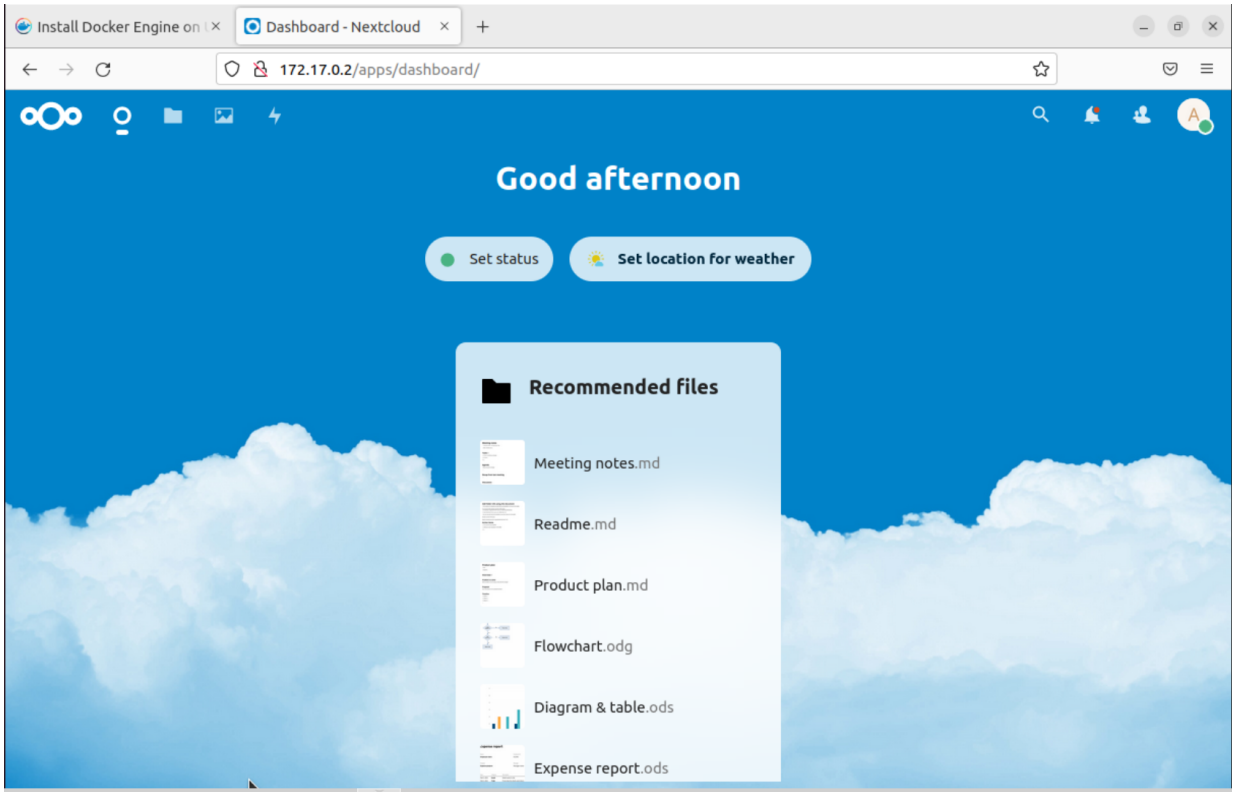
Invocamos el contenedor de nextcloud con el comando.

```
docker inspect <idmaquina>
```


```
            "GlobalIPv6Address": "",
            "GlobalIPv6PrefixLen": 0,
            "IPAddress": "172.17.0.2",
            "IPPrefixLen": 16,
            "IPv6Gateway": "",
            "MacAddress": "02:42:ac:11:00:02",
            "Networks": {
                "bridge": {
                    "IPAMConfig": null,
                    "Links": null,
                    "Aliases": null,
                    "NetworkID": "75abe74206c60f599729093a2d0bd1d01b4fa224ef9cd25b2c132cb128c75a4d",
                    "EndpointID": "f0cc2e48b119e8e6d931a6e58d71c19ac5ef1b9b1a6dc52d121ff4c086ebc941",
                    "Gateway": "172.17.0.1",
                    "IPAddress": "172.17.0.2",
                    "IPPrefixLen": 16,
                    "IPv6Gateway": "",
                    "GlobalIPv6Address": "",
                    "GlobalIPv6PrefixLen": 0,
                    "MacAddress": "02:42:ac:11:00:02",
                    "DriverOpts": null
                }
            }
        }
    }
]
root@pablo-vm:/home/pablo#
```


Comprobamos que la ip de la maquina es 172.17.0.2. Observamos que ha reutilizado la ip de la máquina de nginx ya que esta se encontraba apagada en dicho momento.





Accediendo desde el navegador observamos que nextcloud se ha instalado correctamente.