

3. Instalación paso a paso

Ejecuta los siguientes comandos para instalar **Portainer CE** en tu sistema.

3.1 Crear volumen de datos persistente

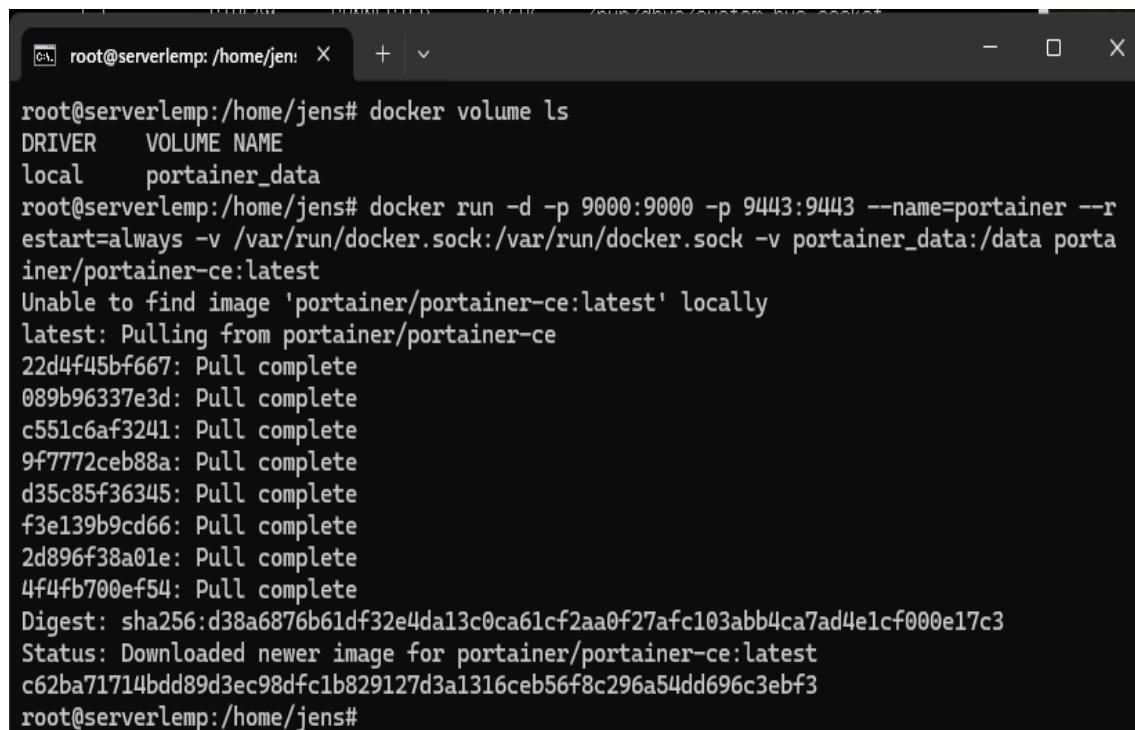
```
sudo docker volume create portainer_data
```

```
root@serverlemp:/home/jens# docker volume create portainer_data
portainer_data
root@serverlemp:/home/jens#
```

Este volumen almacenará la configuración y los datos del servicio.

3.2 Desplegar el contenedor Portainer

```
sudo docker run -d \
-p 9000:9000 \
-p 9443:9443 \
--name=portainer \
--restart=always \
-v /var/run/docker.sock:/var/run/docker.sock \
-v portainer_data:/data \
portainer/portainer-ce:latest
```



The screenshot shows a terminal window with a dark theme. The title bar says "root@serverlemp:/home/jens#". The terminal displays the following command and its execution:

```
root@serverlemp:/home/jens# docker volume ls
DRIVER      VOLUME NAME
local      portainer_data
root@serverlemp:/home/jens# docker run -d -p 9000:9000 -p 9443:9443 --name=portainer --restart=always -v /var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer-ce:latest
Unable to find image 'portainer/portainer-ce:latest' locally
latest: Pulling from portainer/portainer-ce
22d4f45bf667: Pull complete
089b96337e3d: Pull complete
c551c6af3241: Pull complete
9f7772ceb88a: Pull complete
d35c85f36345: Pull complete
f3e139b9cd66: Pull complete
2d896f38a01e: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:d38a6876b61df32e4da13c0ca61cf2aa0f27afc103abb4ca7ad4e1cf000e17c3
Status: Downloaded newer image for portainer/portainer-ce:latest
c62ba71714bdd89d3ec98dfc1b829127d3a1316ceb56f8c296a54dd696c3ebf3
root@serverlemp:/home/jens#
```

Explicación de los parámetros:

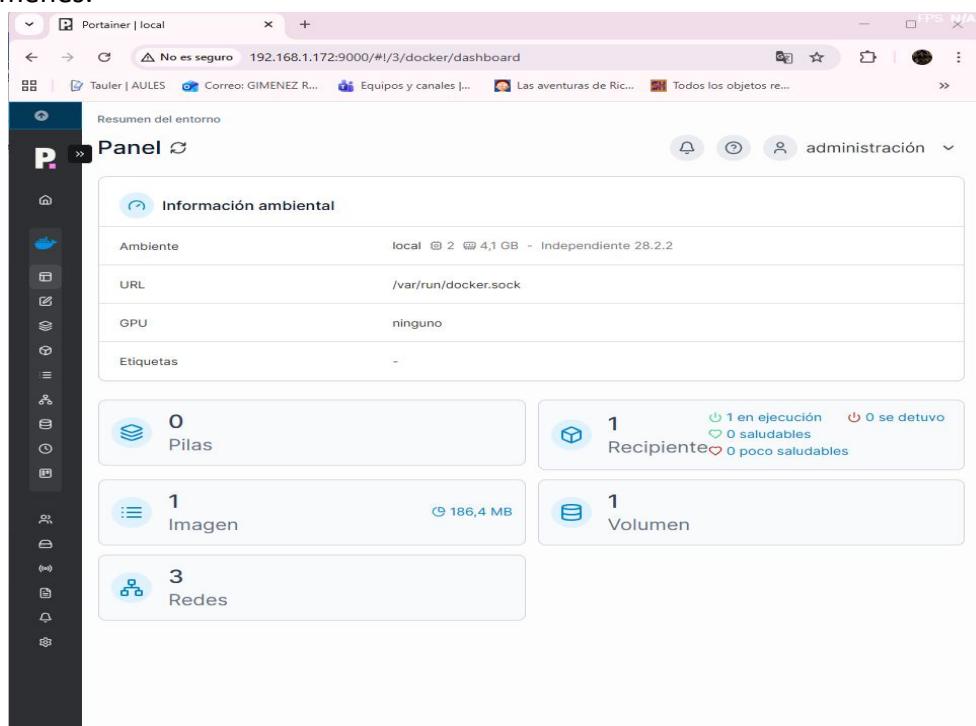
- d → Ejecuta el contenedor en segundo plano.
- p 9000:9000 → Puerto HTTP.
- p 9443:9443 → Puerto HTTPS.
- -restart=always → Reinicia automáticamente al arrancar el sistema.
- v /var/run/docker.sock:/var/run/docker.sock → Permite a Portainer comunicarse con el motor Docker del host.
- v portainer_data:/data → Volumen persistente para configuraciones.

```
root@serverlemp:/home/jens# docker ps
CONTAINER ID   IMAGE          COMMAND       CREATED      STATUS      PORTS     NAMES
c62ba71714bd   portainer/portainer-ce:latest   "/portainer"   31 seconds ago   Up 29 seconds   0.0.0.0:9000->9000/tcp, [::]:9000->9000/tcp, 8000/tcp, 0.0.0.0:9443->9443/tcp, [::]:9443->9443/tcp   portainer
root@serverlemp:/home/jens#
```

4. Acceso y configuración inicial

1. Abre un navegador web desde tu equipo y accede a:
2. http://IP_DEL_SERVIDOR:9000
3. Crea un **usuario administrador** (por ejemplo, admin / admin1234).
4. Selecciona la opción “**Docker local**” para gestionar el entorno Docker de tu máquina.

Una vez dentro, verás el panel principal con acceso a contenedores, imágenes, redes y volúmenes.



5. Exploración de la interfaz

Desde el panel principal se puede acceder a:

- **Containers:** muestra los contenedores en ejecución. Permite ver, detener, eliminar o entrar en la consola del contenedor.
- **Images:** lista las imágenes descargadas en el sistema.
- **Volumes y Networks:** permite ver los volúmenes de datos y redes creadas.
- **Stacks:** permite desplegar servicios definidos mediante archivos docker-compose.yml.

Portainer ofrece una **vista gráfica del estado de los servicios y sus dependencias**, ideal para comprender cómo se comunican entre sí.

6. Gestión y mantenimiento

Para detener o eliminar Portainer:

```
sudo docker stop portainer
```

```
sudo docker rm portainer
```

```
sudo docker volume rm portainer_data
```

El parámetro --restart=always en el comando original garantiza que el contenedor se inicie automáticamente al arrancar el sistema.

7. Apéndice – Comandos Docker útiles

Comando	Descripción
docker ps	Muestra contenedores activos
docker ps -a	Muestra todos los contenedores, incluidos los detenidos
docker images	Lista las imágenes descargadas
docker pull <imagen>	Descarga una imagen del repositorio
docker exec -it <id> bash	Abre consola dentro del contenedor

Comando	Descripción
<code>docker logs <id></code>	Muestra los registros de un contenedor
<code>docker compose up -d</code>	Levanta los servicios definidos en docker-compose.yml
<code>docker compose down</code>	Detiene y elimina los contenedores del compose
<code>docker system prune -a</code>	Limpia imágenes y contenedores no utilizados