

# Guía de instalación y configuración de OMERO-Server y OMERO-Web con Docker y LVM

---

<b>Objetivo</b>	<b>4</b>
<b>Escenario de red asumido</b>	<b>4</b>
<b>Requisitos</b>	<b>4</b>
<b>Identificando los comandos a ejecutar</b>	<b>4</b>
<b>Cambiar a modo administrador (Usuario con privilegios sudo)</b>	<b>6</b>
<b>Instalar LVM (en caso de no instalarlo junto al SO)</b>	<b>6</b>
<b>Preparación del almacenamiento con LVM</b>	<b>6</b>
1. Verifica que se reconozca el disco secundario	6
2. Limpiar disco (opcional, destructivo, borra todo su contenido actual)	7
3. Crear partición LVM	7
4. Crear volumen físico (PV)	7
5. Crear grupo de volúmenes (VG) “omeroStorage”	7
6. Crear volumen lógico (LV)	8
7. Formatear volumen (LV)	8
8. Crear el punto de montaje	8
9. Ubica el UUID	8
10. Editar el archivo /etc/fstab	8
<b>Instalación de Docker</b>	<b>9</b>
1. Crear directorio para contener los archivos necesarios para Omero con Docker	9
2. Agregar llaves GPG oficiales de Docker	9
3. Agregar fuentes de repositorio apt:	10
4. Instalar paquetes docker:	10
5. Verificar la instalación de docker	10
6. Imágenes Docker	10
<b>Preparando el entorno para usar Omero-Server y Omero-Web</b>	<b>11</b>
1. Asegurate de estar en el directorio ~/omero para generar los archivos necesarios para Omero con Docker	11
2. Crear archivo “.env” para variables sensibles	11
3. Crear docker-compose.yml	11
4. Permisos de almacenamiento	13
5. Levantar servicios de Omero	14
6. Verificación del levantamiento de los servicios.	14
7. Conectarse desde el mismo dispositivo.	14

<b>Identificar IP para conectarse desde otro dispositivo</b>	<b>14</b>
1. Asegurate de tener instalado iproute2 gawk	14
2. Para identificar la IP del dispositivo ejecuta:	14
<b>Extiende el espacio de almacenamiento</b>	<b>15</b>
1. Agregando un nuevo disco	15
2. Asegurate de estar en el directorio ~/omero	15
3. Crea el script “extiende_omero.sh”	16
4. Dale permisos de ejecución	17
5. Ejecuta el script	17
<b>Instrucciones para la creación de grupos y nuevos usuarios para el servidor.</b>	<b>18</b>
1. Acceso al servidor	18
2. Creación de grupos de trabajo	18
3. Creacion de usuarios	21
4. Verifica la creación del usuario	22

# Objetivo

Desplegar OMERO Server + OMERO Web en un servidor local, con:

- ❖ Almacenamiento persistente mediante LVM dedicado
- ❖ PostgreSQL aislado (manejado por Docker)
- ❖ Acceso vía red local (LAN)
- ❖ Sin dependencia de:
  - dominio público
  - IP fija
  - acceso a internet externo

## Escenario de red asumido

- ❖ El servidor sólo es accesible dentro de la red local
- ❖ IP asignada por DHCP (IP puede cambiar)
- ❖ Los clientes acceden usando:
  - IP local actual del servidor
  - nombre local (editando `/etc/hosts` del lado del cliente)

## Requisitos

- ❖ Sistema
  - Ubuntu 24.04 LTS
  - Usuario con privilegios `sudo`
  - Acceso a terminal
- ❖ Almacenamiento
  - Un disco separado del sistema operativo (más discos en caso de querer expandir el espacio)
  - LVM disponible
- ❖ Conocimientos mínimos
  - Uso de la línea de comandos en Linux
    - Navegación básica (`cd`, `ls`)
    - Edición de archivos con editores de texto en terminal (por ejemplo, `nano` o `vi`)

## Identificando los comandos a ejecutar

Los comandos que ejecutamos en esta guía los podrás identificar por tener este estilo:

→ comando

Cada comando único se identifica por la flecha antes del inicio del comando en sí mismo. En esta guía te encontrarás comandos separados en múltiples líneas, asegúrate de ejecutarlos como una única línea.

## Identificando contenido de archivos

Los archivos están disponibles en github o puedes crearlos tú mismo.

Esto sería un texto que va en algún archivo como .env, .yml, .sh, etc.

En caso de copiar y pegar directamente de este documento, hay que mantener el formato (indentaciones) mostrado aquí para que el archivo sea funcional.

---

## Cambiar a modo administrador (Usuario con privilegios sudo)

Ejecuta el comando `sudo su` para entrar en modo administrador.

→ `sudo su`

En caso de no hacer esto deberás agregar `sudo` al inicio de todos los comandos usados en esta guía.

---

## Instalar LVM (en caso de no instalarlo junto al SO)

→ `apt update`  
→ `apt install -y lvm2`

---

## Preparación del almacenamiento con LVM

1. Verifica que se reconozca el disco secundario  
→ `lsblk -o NAME,SIZE,TYPE,MOUNTPOINT`

o

→ `lsblk -o NAME #para ver únicamente los nombres`

En esta guía se asume que el nuevo disco es: `/dev/sdb`.

En caso de que el disco nuevo no sea `sdb` cambiar el nombre en las siguientes instrucciones.

El nombre del disco se puede averiguar si se ejecuta el comando antes y después de haber conectado el disco secundario y comparar los resultados del comando.



(Nota: en el escenario donde se instaló LVM en un SO preexistente la línea `ubuntu -- vg - ubuntu -- lv` no aparecerá, esto no afecta al proceso descrito en esta guía).

## 2. Limpiar disco (opcional, destructivo, borra todo su contenido actual)

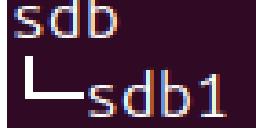
→ `wipefs -a /dev/sdb`

## 3. Crear partición LVM

→ `parted /dev/sdb --script mklabel gpt`  
 → `parted /dev/sdb --script mkpart primary 0% 100%`  
 → `parted /dev/sdb --script set 1 lvm on`

## 4. Crear volumen físico (PV)

→ `pvcreate /dev/sdb1`



(Ejecutando el comando del paso 1 de esta sección podrás ver cómo se generó el PV).

## 5. Crear grupo de volúmenes (VG) “omeroStorage”

→ `vgcreate omeroStorage /dev/sdb1`

## 6. Crear volumen lógico (LV)

→ `lvcreate -l 100%FREE -n omeroData omeroStorage`

```
sdb
└─sdb1
    └─omeroStorage - omeroData
```

(Ejecutando el comando del paso [1](#) de esta sección podrás ver cómo se generó el LV).

## 7. Formatear volumen (LV)

→ `mkfs.ext4 /dev/omeroStorage/omeroData`

## 8. Crear el punto de montaje

→ `mkdir -p /omero/data`

## 9. Ubica el UUID

→ `blkid /dev/omeroStorage/omeroData`

Al ejecutar el comando se obtendrá una secuencia de alfanumérica que es el Identificador único universal ([Universally unique identifier o UUID](#))  
**(debes preservar este dato ya que será necesario en el siguiente paso)**

**UUID="f4427e5f-4f4f-4734-8f9a-ae31eea8124c"**

Esta imagen es un ejemplo de un posible UUID.

## 10. Editar el archivo `/etc/fstab`

Puedes ejecutar el comando nano `/etc/fstab` para editarlo.

Reemplaza `<EL_UUID_AQUI>` por el UUID que obtuviste previamente.

→ `UUID=<EL_UUID_AQUI> /omero/data ext4 defaults 0 2`

Agrega la línea de texto al archivo `/etc/fstab` y guarda los cambios.

---

```
GNU nano 7.2                                     /etc/fstab *
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>  <type>  <options>      <dump>  <pass>
# / was on /dev/ubuntu-vg/ubuntu-lv during curtin installation
/dev/disk/by-id/dm-uuid-LVM-qb8nACe3hZ2I2JbAiABkUg0IQGg6R0nsQU8qq0R10YOKLlVKZNI
# /boot was on /dev/sda2 during curtin installation
/dev/disk/by-uuid/cc92fbf5-1d46-4976-ad91-3d862257cbd9 /boot ext4 defaults 0 1
/swap.img      none    swap    sw     0      0
UUID=f4427e5f-4f4f-4734-8f9a-ae31eea8124c /omero/data ext4 defaults 0 2
```

Esta imagen es un ejemplo de cómo debe verse el archivo `/etc/fstab` una vez agregues la línea indicada con tu UUID.

#### 11. Montar el disco:

- `systemctl daemon-reload`
  - `mount -a`
- 

## Instalación de Docker

### 1. Crear directorio para contener los archivos necesarios para Omero con Docker

- `mkdir -p ~/omero`
- `cd ~/omero`

### 2. Agregar llaves GPG oficiales de Docker

- `apt remove $(dpkg --get-selections docker.io docker-compose docker-compose-v2 docker-doc podman-docker containerd runc | cut -f1)`
- `apt update`
- `apt install -y ca-certificates curl`
- `install -m 0755 -d /etc/apt/keyrings`

```
→ curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc  
→ chmod a+r /etc/apt/keyrings/docker.asc
```

### 3. Agregar fuentes de repositorio apt:

```
→ tee /etc/apt/sources.list.d/docker.sources <<EOF  
→ Types: deb  
→ URLs: https://download.docker.com/linux/ubuntu  
→ Suites: $(. /etc/os-release && echo "$UBUNTU_CODENAME")  
→ Components: stable  
→ Signed-By: /etc/apt/keyrings/docker.asc  
→ EOF  
→ apt update
```

### 4. Instalar paquetes docker:

```
→ apt install -y docker-ce docker-ce-cli containerd.io  
docker-buildx-plugin docker-compose-plugin
```

### 5. Verificar la instalación de docker

```
→ systemctl status docker
```

```
● docker.service - Docker Application Container Engine  
  Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled; preset: enabled)  
  Active: active (running) since Sun 2025-11-23 22:25:03 CST; 2h 26min ago  
TriggeredBy: ● docker.socket  
    Docs: https://docs.docker.com  
   Main PID: 1428 (dockerd)
```

### 6. Imágenes Docker

```
→ docker pull postgres  
→ docker pull openmicroscopy/omero-server  
→ docker pull openmicroscopy/omero-web-standalone
```

# Preparando el entorno para usar Omero-Server y Omero-Web

1. Asegurate de estar en el directorio `~/omero` para generar los archivos necesarios para Omero con Docker

→ `cd ~/omero`

2. Crear archivo “`.env`” para variables sensibles

→ `nano .env`

Contenido del archivo `.env`:

```
# --- Base de datos ---
OMERO_DB_USER=omero_db
OMERO_DB_PASS=CAMBIAR_PASSWORD
OMERO_DB_NAME=omero
# --- Servidor OMERO ---
OMERO_ROOT_PASS=CAMBIAR_ROOT_PASS
# --- Web ---
OMERO_WEB_PORT=8080:8080
```

3. Crear `docker-compose.yml`

Puedes descargar el archivo de:

<https://github.com/RPintoC/omero-lvm-docker-guide/blob/main/docker-compose.yml>

O puedes crearlo tu mismo:

→ `nano docker-compose.yml`

Contenido del archivo `docker-compose.yml` (la indentación debe recrearse exactamente como se ven en este documento para que funcione, la indentación la generas con la tecla Tab):

```
services:
  db:
    image: postgres:13
    container_name: omero-db
```

```
restart: always
env_file: .env
environment:
  POSTGRES_USER: ${OMERO_DB_USER}
  POSTGRES_PASSWORD: ${OMERO_DB_PASS}
  POSTGRES_DB: ${OMERO_DB_NAME}
volumes:
  - omero-db:/var/lib/postgresql/data
shm_size: 1g
healthcheck:
  test: ["CMD-SHELL", "pg_isready -U ${OMERO_DB_USER}"]
  interval: 10s
  timeout: 5s
  retries: 5
networks:
  - omero-net
server:
  image: openmicroscopy/omero-server
  container_name: omero-server
  restart: always
  env_file: .env
  environment:
    CONFIG_omero_db_host: omero-db
    CONFIG_omero_db_user: ${OMERO_DB_USER}
    CONFIG_omero_db_pass: ${OMERO_DB_PASS}
    CONFIG_omero_db_name: ${OMERO_DB_NAME}
    ROOTPASS: ${OMERO_ROOT_PASS}
  volumes:
    - /omero/data:/OMERO
depends_on:
  db:
    condition: service_healthy
networks:
  - omero-net
web:
  image: openmicroscopy/omero-web-standalone
  container_name: omero-web
  restart: always
  env_file: .env
  environment:
    OMEROHOST: omero-server
  ports:
    - "${OMERO_WEB_PORT}"
```

```
depends_on:
  - server
networks:
  - omero-net
networks:
  omero-net:
volumes:
  omero-db:
```

Debe verse como la siguiente imagen (la imagen no muestra todo el contenido del archivo).

```
GNU nano 7.2                                            docker-compose.y
services:
  db:
    image: postgres:13
    container_name: omero-db
    restart: always
    env_file: .env
    environment:
      POSTGRES_USER: ${OMERO_DB_USER}
      POSTGRES_PASSWORD: ${OMERO_DB_PASS}
      POSTGRES_DB: ${OMERO_DB_NAME}
    volumes:
      - omero-db:/var/lib/postgresql/data
    shm_size: 1g
    healthcheck:
      test: ["CMD-SHELL", "pg_isready -U ${OMERO_DB_USER}"]
      interval: 10s
      timeout: 5s
      retries: 5
    networks:
      - omero-net
```

#### 4. Permisos de almacenamiento

Antes de levantar los servicios hay que asignar los permisos a docker para acceder a omero/Data en omeroStorage.

- chown -R 1000:1000 /omero/data
- chmod -R 775 /omero/data

## 5. Levantar servicios de Omero

Una vez levantados satisfactoriamente, se iniciaran incluso en caso de resetear la máquina.

→ docker compose up -d

## 6. Verificación del levantamiento de los servicios.

→ docker compose ps

## 7. Conectarse desde el mismo dispositivo.

Abre cualquier navegador y accede a la siguiente URL:

localhost:4080

---

## Identificar IP para conectarse desde otro dispositivo

### 1. Asegurate de tener instalado iproute2 gawk

Para instalarlo ejecuta:

→ sudo apt update  
→ sudo apt install iproute2 gawk

### 2. Para identificar la IP del dispositivo ejecuta:

→ ip route show default

o para obtener la ip específica:

→ ip route show default | awk '{print \$9}'

```
root@ocean:~# ip route show default
default via 192.168.0.1 dev enp0s3 proto dhcp src 192.168.0.6 metric 100
root@ocean:~# ip route show default | awk '{print $9}'
192.168.0.6
```

Ejemplo de conexión al servidor mediante IP en navegador (usa la ip de tu dispositivo)

`http://<IP>:4080`

`http://192.168.0.6:4080`

---

## Extiende el espacio de almacenamiento

### 1. Agregando un nuevo disco

Conecta un nuevo disco y verificar que se reconozca (usualmente “`sdc`” en caso contrario reemplazar el nombre en las siguientes instrucciones)

→ `lsblk -o NAME,SIZE,TYPE,MOUNTPOINT`

o

→ `lsblk -o NAME #para ver únicamente los nombres`

```
sda
└─sda1
└─sda2
└─sda3
  └─ubuntu--vg-ubuntu--lv
sdb
└─sdb1
  └─omeroStorage-omeroData
sdc
```

### 2. Asegurate de estar en el directorio `~/omero`

→ `cd ~/omero`

### 3. Crea el script “extiende\_omero.sh”

Puedes descargar el archivo de:

[https://github.com/RPintoC/omero-lvm-docker-guide/blob/main/extiende\\_omero.sh](https://github.com/RPintoC/omero-lvm-docker-guide/blob/main/extiende_omero.sh)

O puedes crearlo tu mismo:

→ **nano extiende\_omero.sh**

Contenido del archivo **extiende\_omero.sh** (la indentación debe recrearse exactamente como se ven en este documento para que funcione, la indentación la generas con la tecla Tab):

```
#!/bin/bash
# extiende_omero.sh
# Uso: ./extiende_omero.sh sdc
set -e
if [ "$EUID" -ne 0 ]; then
    echo "Ejecuta este script como root"
    exit 1
fi
if [ -z "$1" ]; then
    echo "Uso: $0 <nombre_del_disco> (ej. sdc)"
    exit 1
fi
DISCO="/dev/$1"
VG="omeroStorage"
LV="omeroData"
MOUNT_POINT="/omero/data"
# Verifica que el disco exista
if [ ! -b "$DISCO" ]; then
    echo "Error: $DISCO no existe"
    exit 1
fi
echo "Creando PV en $DISCO..."
pvcreate "$DISCO"
echo "Agregando $DISCO al VG $VG..."
vgextend "$VG" "$DISCO"
echo "Extendiendo LV $LV al máximo disponible..."
lvextend -l +100%FREE "/dev/$VG/$LV"
echo "Redimensionando filesystem (ext4) en $MOUNT_POINT..."
resize2fs "/dev/$VG/$LV"
echo "¡Listo! Nuevo disco $DISCO agregado a $VG/$LV y filesystem
```

redimensionado."  
df -h "\$MOUNT\_POINT"

4. Dale permisos de ejecución

→ chmod +x extiende\_omero.sh

5. Ejecuta el script

→ ./extiende\_omero.sh sdc

# Instrucciones para la creación de grupos y nuevos usuarios para el servidor.

## 1. Acceso al servidor

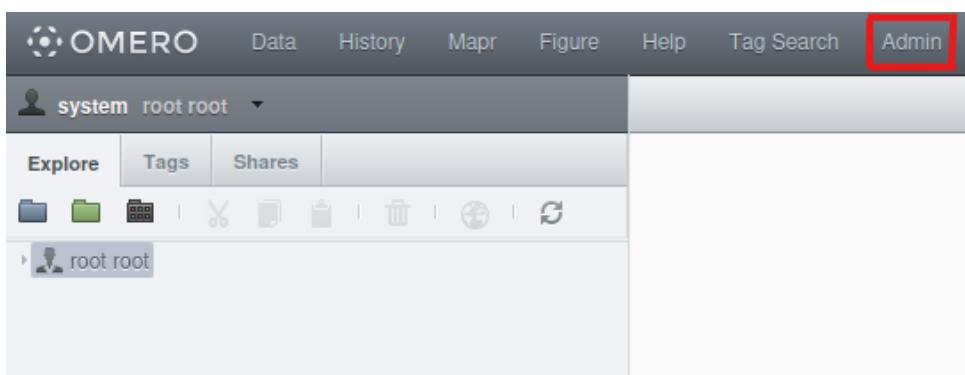
Abre un navegador web y accede a OMERO usando la IP del servidor o localhost si estás trabajando desde la máquina host.

Inicia sesión con las credenciales de administrador (root).



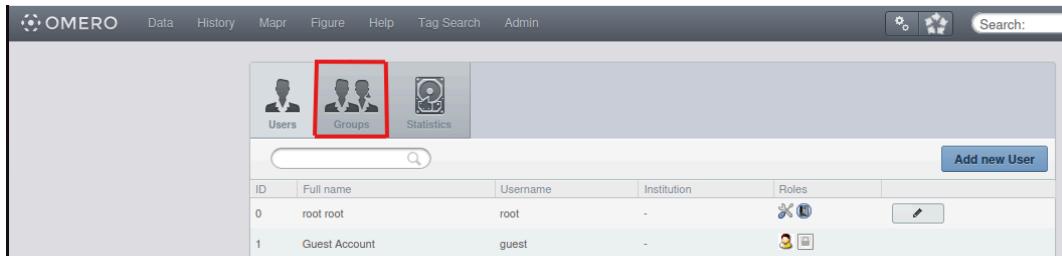
## 2. Creación de grupos de trabajo

Una vez dentro de OMERO, en la barra superior, selecciona la opción “Admin”. Esto te llevará a la sección de gestión de usuarios por defecto.



Para crear un grupo (necesario si aún no existe ninguno):

Selecciona la pestaña “Groups”.



The screenshot shows the OMERO web interface. At the top, there is a navigation bar with links for Data, History, Mapr, Figure, Help, Tag Search, and Admin. Below the navigation bar is a toolbar with icons for search, refresh, and user management. The main content area has three tabs: 'Users', 'Groups' (which is highlighted with a red box), and 'Statistics'. A search bar is located below the tabs. A table lists users with columns for ID, Full name, Username, Institution, and Roles. There are buttons for 'Add new User' and edit/refresh icons for each row. The table data is as follows:

ID	Full name	Username	Institution	Roles
0	root root	root	-	Administrator
1	Guest Account	guest	-	Guest

Aquí se listan todos los grupos existentes. Para crear uno nuevo, haz clic en “Add new Group”.



This screenshot shows the 'Groups' tab selected in the OMERO interface. It features a search bar and an 'Add new Group' button highlighted with a red box. Below these are three groups listed in a table:

ID	Group Name	Permission	Description
0	system	Private (rw----	-
1	user	Read-Only (rwr-r-)	-
2	guest	Private (rw----	-

Rellena la información requerida para el grupo (nombre, descripción, etc.).

Guarda los cambios.

## Add group

[Cancel](#)

Fields marked in red are mandatory.

Name:

Description:

Owners:  Type owner names to

Members:  Type member names

Permissions:

- Private
- Read-Only
- Read-Annotate
- Read-Write

### 3. Creacion de usuarios

Vuelve a la pestaña “Users” dentro de la sección Admin.

Haz clic en “Add new User”.

ID	Full name	Username	Institution	Roles	
0	root root	root	-		
1	Guest Account	guest	-		

Completa los campos obligatorios:

- Nombre de usuario
- Contraseña
- Otros campos marcados en rojo.

New User Cancel

Fields marked in red are mandatory.

Username:

Password:

Confirmation:

First name:

Middle name:

Last name:

Email:

Institution:

Role:  
 User  
 Administrator  
 Administrator with restricted privileges

Active:

Groups:

Default group:

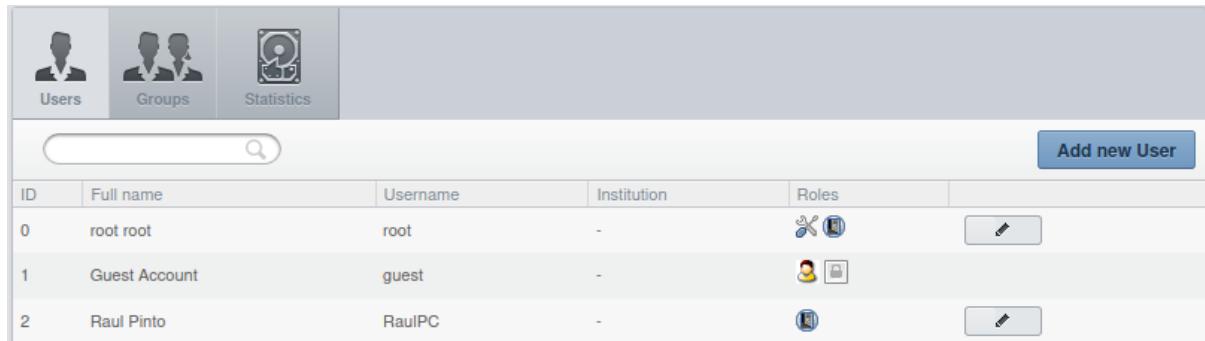
Save

En la sección “Groups”, selecciona el grupo al que pertenecerá el usuario (puede ser el grupo recién creado o uno existente).

Guarda los cambios.

#### 4. Verifica la creación del usuario

Con esto habrás habilitado a tu primer usuario en el servidor, podrás verificarlo en la pestaña “Users”.



The screenshot shows a user management interface with the following components:

- Top navigation bar:** Three tabs: "Users" (selected), "Groups", and "Statistics".
- Search bar:** A search input field with a magnifying glass icon.
- Add new User button:** A blue button labeled "Add new User" with a pencil icon.
- User table:** A table listing three users with columns: ID, Full name, Username, Institution, and Roles.

ID	Full name	Username	Institution	Roles	
0	root root	root	-	🔧 🔑	
1	Guest Account	guest	-	👤 🔒	
2	Raul Pinto	RaulPC	-	👤	