

Immich

1. Introducción

Immich es una solución de gestión de fotos y vídeos de alto rendimiento, diseñada como alternativa autohospedada a Google Photos. Incluye reconocimiento facial, geolocalización y transcodificación de vídeo.

Debido a la gran cantidad de datos que genera (fotos originales, miniaturas, vídeos y base de datos vectorial para IA), el almacenamiento es el componente crítico de este despliegue.

2. Arquitectura de Almacenamiento (RAID 01)

Para gestionar los datos, utilizamos LVM (Logical Volume Manager). A diferencia de las particiones tradicionales estáticas, LVM nos permite:

1. Agrupar varios discos físicos en un solo "Grupo de Volúmenes" (VG).
2. Crear "Volúmenes Lógicos" (LV) que podemos redimensionar fácilmente.
3. Expandir el espacio en el futuro añadiendo un disco nuevo sin necesidad de mover los datos ni reinstalar.

Configuración realizada:

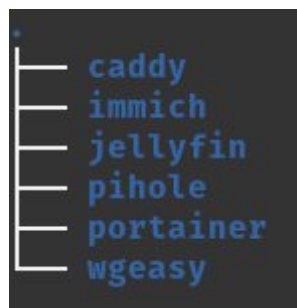
- Punto de Montaje: Se ha montado el volumen lógico principal en una ruta del sistema host, la cual pasaremos al contenedor.

Comando para verificar la estructura LVM y el punto de montaje: "lsblk -f" (Este comando muestra el árbol de discos y confirma que estamos usando lvm)

3. Estructura de Directorios

Preparamos la carpeta de trabajo donde alojaremos los archivos de configuración.

Comando a ejecutar: "mkdir -p ~/docker/immich"



4. Configuración de Seguridad (UFW)

Immich escucha por defecto en el puerto 2283. Debemos abrirlo para permitir el acceso desde la red local o para que el Proxy Inverso pueda conectar.

Comando a ejecutar: "sudo ufw allow 2283/tcp comment 'Immich Web UI'"

2283/tcp	ALLOW	Anywhere	# Immich
----------	-------	----------	----------

5. Configuración del Entorno (.env)

Immich requiere un archivo de variables de entorno para definir las contraseñas y, **muy importante**, la ubicación de las fotos en nuestro RAID.

Creamos el archivo `.env` en la carpeta `~/docker/immich` con el siguiente contenido.

Nota: En la variable `UPLOAD_LOCATION` definimos la ruta a nuestro punto de montaje RAID.

Código del archivo `.env`: `"# BASE IMMICH_VERSION=release`

Ajusta esta ruta a tu punto de montaje real

`UPLOAD_LOCATION=/mnt/mi RAID/immich_data`

```
# The location where your uploaded files are stored
UPLOAD_LOCATION=/home/eros/docker/immich/datos/galeria
```

`DB_DATA_LOCATION=./postgres`

```
# The location where your database files are stored. Network sha
DB_DATA_LOCATION=/home/eros/docker/immich/datos/bbdd/postgres
```

CONTRASEÑAS (Cambiar por seguridad)

`DB_PASSWORD=postgres DB_USERNAME=postgres DB_DATABASE_NAME=immich"`

6. Archivo de Despliegue (docker-compose.yml)

Para levantar el servicio, utilizaremos el archivo `docker-compose.yml` adjunto a esta documentación (basado en la versión oficial recomendada). Este archivo leerá las rutas definidas anteriormente.

```
#
# The compose file on main may not be compatible with the latest release.

name: immich

# Run All Services
services:
  # Run Service
  immich-server:
    container_name: immich_server
    image: ghcr.io/immich-app/immich-server:${IMMICH_VERSION:-release}
    # extends:
    #   file: hwaccel.transcoding.yml
    #   service: cpu # set to one of [nvcv, quicksync, rkmp, vaapi, vaapi-wsl] for accelerated transcoding
    volumes:
      # Do not edit the next line. If you want to change the media storage location on your system, edit the value of UPLOAD_LOCATION
      - $(UPLOAD_LOCATION):/data
      - /etc/localtime:/etc/localtime:ro
    env_file:
      - .env
    ports:
      - '2283:2283'
    depends_on:
      - redis
      - database
    restart: always
    healthcheck:
      disable: false

  # Run Service
  immich-machine-learning:
    container_name: immich_machine_learning
    # For hardware acceleration, add one of -[armnn, cuda, rocm, openvino, rknn] to the image tag.
    # Example tag: $(IMMICH_VERSION:-release)-cuda
    image: ghcr.io/immich-app/immich-machine-learning:${IMMICH_VERSION:-release}
    # extends: # uncomment this section for hardware acceleration - see https://docs.immich.app/features/ml-hardware-acceleration
    #   file: hwaccel.ml.yml
    #   service: cpu # set to one of [armnn, cuda, rocm, openvino, openvino-wsl, rknn] for accelerated inference - use the '...' tag
    volumes:
      - model-cache:/cache
    env_file:
      - .env
    restart: always
    healthcheck:
      disable: false

  # Run Service
  redis:
    container_name: immich_redis
    image: docker.io/valkey/valkey:8-bookworm@sha256:fea8b3e67b15729d4bb70589eb03367bab9ad1ee89c876f54327fc7c6e618571
    healthcheck:
      test: redis-cli ping || exit 1
    restart: always

  # Run Service
  database:
    container_name: immich_postgres
    image: ghcr.io/immich-app/postgres:14-vectorchord@sha256:b6f63357191b76a916ae5eb93464d65c07511da41e3bf
    environment:
      POSTGRES_PASSWORD: ${DB_PASSWORD}
      POSTGRES_USER: ${DB_USERNAME}
      POSTGRES_DB: ${DB_DATABASE_NAME}
      POSTGRES_INITDB_ARGS: '--data-checksums'
      # Uncomment the DB_STORAGE_TYPE: 'HDD' var if your database isn't stored on SSDs
      # DB_STORAGE_TYPE: 'HDD'
    volumes:
      # Do not edit the next line. If you want to change the database storage location on your system, edit the value of DB_DATA_LOCATION
      - $(DB_DATA_LOCATION):/var/lib/postgresql/data
      shm_size: 128mb
    restart: always

volumes:
  model-cache:
```

7. Integración con Caddy (Acceso Externo)

Para acceder a nuestras fotos desde el móvil fuera de casa, añadimos la entrada al Caddyfile.

Código a añadir en Caddyfile: "fotos.tu-subdominio.duckdns.org { reverse_proxy 127.0.0.1:2283 }"

```
#Immich
immich [redacted] org {
    reverse_proxy [redacted] :2283
}
```

8. Despliegue y Verificación

Levantamos el stack completo (Servidor, Microservicios, Machine Learning, Redis y Postgres).

Comando a ejecutar: "docker compose up -d"

```

    portainer
562ace11b087 ghcr.io/immich-app/immich-server:release "tini -- /bin
/bash -..." 3 weeks ago Up 2 days (healthy) 0.0.0.0:2283->2283/tcp, [::]:2283->2283/tcp

    immich_server
d66a1a564765 ghcr.io/immich-app/immich-machine-learning:release "tini -- pyth
on -m i..." 3 weeks ago Up 2 days (healthy)

    immich_machine_learning
170135355a14 ghcr.io/immich-app/postgres:14-vectorchord0.4.3-pgvector0.2.0 "/usr/local/b
in/immich..." 3 weeks ago Up 2 days (healthy) 5432/tcp

```

9. Acceso Inicial

Accedemos vía web a <http://localhost:2283> (o nuestro dominio). La primera vez nos pedirá crear la cuenta de administrador.

