

Jellyfin

1. Introducción

Jellyfin es un sistema de medios que nos permite organizar y transmitir nuestras películas y series. En este despliegue configuraremos la **Aceleración por Hardware** para utilizar la gráfica integrada en el procesador (iGPU). Esto es vital para que, si necesitamos ver una película en el móvil con menor calidad, sea el procesador gráfico quien haga el trabajo pesado y no se sature la CPU.

2. Configuración de Almacenamiento

En este caso, utilizaremos una estructura de carpetas estándar en el disco local para alojar nuestra biblioteca multimedia.

Comando a ejecutar (Crear carpetas de configuración y multimedia):

```
"mkdir -p ~/docker/jellyfin/config ~/docker/jellyfin/cache"
"mkdir -p ~/docker/jellyfin/media"
```

(Nota: Si ya tienes tus películas en otra carpeta como ~/Videos, usaremos esa ruta en el archivo docker-compose)



3. Verificación de Hardware (iGPU)

Antes de desplegar, verificamos que el sistema detecta la gráfica integrada del procesador. Buscamos la carpeta `dri` (Direct Rendering Infrastructure).

Comando a ejecutar: "ls -l /dev/dri" (*Deberíamos ver card0 y renderD128. renderD128 es el dispositivo que usará Jellyfin*)

```

total 0
drwxr-xr-x 2 root root          80 dic 14 17:31 by-path
crw-rw---- 1 root video 226,   1 dic 14 17:31 card1
crw-rw---- 1 root render 226, 128 dic 14 17:31 renderD128

```

4. Configuración de Seguridad (UFW)

Abrimos el puerto web de Jellyfin.

Comando a ejecutar: "sudo ufw allow 8096/tcp comment 'Jellyfin Media Server'"

8096/tcp	ALLOW	Anywhere	# Jellyfin
----------	-------	----------	------------

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

Creamos el archivo docker-compose.yml.

Cambios importantes:

- **devices:** Mapeamos /dev/dri para que el contenedor vea la iGPU.
- **volumes:** Ajustamos la ruta de /media a la carpeta donde tengas tus archivos reales.

```
services:
  jellyfin:
    image: jellyfin/jellyfin:latest
    container_name: jellyfin
    user: "${UID:-1000}:${GID:-1000}"
    network_mode: "host"
    volumes:
      - jellyfin_config:/config
      - jellyfin_cache:/cache
      - /home/eros/media:/media
    restart: unless-stopped

volumes:
  jellyfin_config:
  jellyfin_cache:
```

6. Integración con Caddy (Acceso Externo)

Configuramos el Proxy Inverso para acceder a Jellyfin desde fuera usando nuestro dominio.

Código a añadir en Caddyfile: "tv.tu-subdominio.duckdns.org { reverse_proxy 127.0.0.1:8096 }"

```
#Jellyfin
jelly[REDACTED].org {
  reverse_proxy [REDACTED]:8096
}
```

7. Despliegue y Verificación

Levantamos el servicio.

Comando a ejecutar: "docker compose up -d"

```
bc3823dd85ce  jellyfin/jellyfin:latest
llyfin"      7 weeks ago   Up 2 days (healthy)

jellyfin
86566b25410e  caddy:latest
```

Configuración de Transcodificación

Una vez dentro del panel de Jellyfin (<http://localhost:8096>), debemos ir a **Panel de Control > Reproducción** y seleccionar:

- **Aceleración por Hardware:** VAAPI (Intel/AMD) o Intel QuickSync (si es Intel moderno).
- **Dispositivo:** /dev/dri/renderD128

