

jellyfin

ASIX2

Beatriz Suarez Luna, Joel Diaz, Marc Mountoto y Leonardo
Duarte
ALINA GARCIA

¿Qué es Jellyfin?

Jellyfin es un **servidor multimedia de código abierto** que permite organizar, gestionar y acceder a contenido como películas, series, música y televisión desde cualquier dispositivo dentro de una red local. Además, como veremos en la parte práctica, también puede configurarse para acceso remoto mediante **reenvío de puertos o túneles**, lo que facilita su uso desde cualquier ubicación fuera de la red local.

Una de sus mayores ventajas es su **accesibilidad multiplataforma**, ya que ofrece tanto una **interfaz web** como **aplicaciones nativas**, lo que lo hace ideal para su uso en móviles, tablets, ordenadores y televisores inteligentes.

A diferencia de otras plataformas como Plex o Emby, Jellyfin es **completamente gratuito** y no limita funcionalidades tras un pago, lo que lo convierte en una excelente alternativa para usuarios que buscan **flexibilidad y personalización** sin costes adicionales.



🔍 Características principales de Jellyfin

📌 Característica	💡 Descripción detallada	💡 Beneficio clave
Servidor multimedia centralizado	Almacena y organiza películas, series, música y fotos en una biblioteca unificada.	Todo tu contenido en un solo lugar, fácil de gestionar.
Acceso multiplataforma	Soporte para Windows, macOS, Linux, Android, iOS, Smart TVs,	Acceso desde cualquier

 Característica	 Descripción detallada	 Beneficio clave
	Apple TV, Roku, Fire TV y navegadores web.	dispositivo, en cualquier lugar.
 Transmisión remota	Configurable para acceso externo seguro mediante VPN, <i>reenvío de puertos o tunneling</i> .	Disfruta tu contenido incluso fuera de casa.
 Código abierto (Open Source)	Licencia libre (GPL), modificable y sin restricciones de pago.	Sin costes ocultos, transparente y personalizable.
 Soporte para plugins	Amplía funciones con <i>plugins oficiales y de la comunidad</i> (subtítulos, metadata, skins).	Adapta Jellyfin a tus necesidades específicas.

Ventajas de Jellyfin

Jellyfin destaca por ser una solución **100% gratuita y de código abierto**, lo que permite a los usuarios evitar costes de suscripción y tener control total sobre su servidor multimedia. A diferencia de alternativas como Plex o Emby, **no hay funciones premium bloqueadas**, ofreciendo acceso completo desde el primer momento. Su enfoque en la **privacidad** es otro punto fuerte, ya que al ser autohospedado, garantiza que tus archivos multimedia no sean analizados ni compartidos con terceros. Además, su compatibilidad con **múltiples plataformas** (Windows, Linux, Android, iOS, Smart TVs, etc.) lo hace accesible desde casi cualquier dispositivo. Para usuarios avanzados, la posibilidad de **personalizar el código y usar plugins** lo convierte en una herramienta muy flexible.

Ventajas y desventajas de Jellyfin

Ventajas

Característica	Explicación	Beneficio clave
 Código abierto y gratuito	No tiene licencias de pago ni funciones bloqueadas.	Ahorro de costos y libertad para usar todas las funciones sin restricciones.
 Privacidad y control	Tus datos no son recolectados por terceros, ya que el servidor es autohospedado.	Mayor seguridad y confianza al manejar tu propia biblioteca multimedia.
 Multiplataforma	Soporta Windows, Linux, macOS, Android, iOS, Smart TVs y más.	Accesible desde casi cualquier dispositivo.
 Personalización avanzada	Permite modificar el código, instalar plugins y adaptar la interfaz.	Flexibilidad para ajustar el servidor a tus necesidades específicas.
 Sin dependencia de la nube	Todo se almacena y gestiona localmente.	No hay riesgo

◆ Desventajas de Jellyfin

Sin embargo, Jellyfin aún tiene áreas de mejora. Su **interfaz gráfica**, aunque funcional, no está tan pulida como la de Plex o Emby, lo que puede afectar a la experiencia de usuario, especialmente para principiantes. En cuanto a **rendimiento**, la transcodificación de medios (convertir videos en tiempo real para adaptarlos a distintos dispositivos) puede ser menos eficiente en hardware modesto, a menos que se use una GPU dedicada. También hay que considerar que **algunos dispositivos** (como ciertas Smart TVs o consolas) pueden requerir

configuraciones adicionales para funcionar correctamente, ya que Jellyfin no tiene el mismo soporte universal que sus competidores. Por último, aunque su sistema de **plugins** es útil, todavía no alcanza la variedad y madurez de las plataformas de pago.

Desventajas

Aspecto	Limitación	Comparación con alternativas
 Interfaz menos pulida	Su diseño es funcional pero no tan refinado como el de Plex o Emby.	Puede requerir más tiempo de adaptación para usuarios nuevos.
 Transcodificación limitada	Rendimiento inferior en hardware modesto, especialmente sin GPU dedicada.	Plex y Emby optimizan mejor la conversión de formatos en tiempo real.
 Compatibilidad reducida	Algunos dispositivos (como ciertas Smart TVs) pueden necesitar configuraciones adicionales.	Plex tiene soporte más amplio en dispositivos comerciales.
 Plugins menos maduros	Su ecosistema de complementos es más pequeño que el de la competencia.	Menor variedad de integraciones avanzadas (ej.: servicios de metadata)

JELLYFIN VS PLEX

Jellyfin y Plex son dos de las plataformas de servidores multimedia más populares, pero con enfoques distintos. Jellyfin destaca por ser completamente gratuito y de código abierto, ofreciendo total control y privacidad al usuario, ya que todo se gestiona localmente sin depender de servidores externos. Sin embargo, requiere más configuración manual y su interfaz, aunque funcional, no está tan pulida como la de Plex. Por otro lado, Plex ofrece una experiencia más refinada y fácil de usar, con soporte nativo para una amplia gama de dispositivos y funciones

avanzadas como transcodificación optimizada y grabación de TV, aunque muchas de estas características requieren una suscripción Plex Pass. Además, Plex depende parcialmente de sus servidores para autenticación y metadatos, lo que puede plantear preocupaciones de privacidad para algunos usuarios. En resumen, si valoras la privacidad y la gratuidad, Jellyfin es la mejor opción; si prefieres comodidad y funciones avanzadas, Plex resulta más conveniente, aunque con posibles costes asociados.

🔍 Comparativa detallada: Jellyfin vs Plex

Criterio	Jellyfin	Plex	¿Cuál gana?
👉 Modelo de negocio	100% gratuito y open-source (sin funciones bloqueadas).	Freemium (versión base gratuita, pero funciones avanzadas requieren Plex Pass).	<input checked="" type="checkbox"/> Jellyfin (sin costes ocultos).
🎯 Facilidad de uso	Interfaz funcional pero menos pulida. Requiere más configuración manual.	Interfaz intuitiva y optimizada. Configuración casi automática.	<input checked="" type="checkbox"/> Plex (mejor para principiantes).
⚙️ Funcionalidades	Buenas funciones básicas. Plugins de comunidad. Falta madurez en opciones avanzadas.	Transcodificación avanzada, DVR, servidor de música, apps móviles premium.	<input checked="" type="checkbox"/> Plex (más completo con Plex Pass).

Criterio	Jellyfin 	Plex 	¿Cuál gana? 
 Privacidad	Todo local (sin nube). Cero recopilación de datos.	Usa servidores de Plex para autenticación/metadatos. Posible tracking de uso.	 Jellyfin (ideal para privacidad).
 Compatibilidad	Soporta muchas plataformas, pero puede necesitar ajustes (ej.: Smart TVs antiguas).	Amplio soporte oficial (apps nativas en consolas, TVs, móviles).	 Plex (más "plug-and-play").
 Transcodificación	Funciona, pero depende mucho del hardware . Sin optimizaciones exclusivas.	Hardware-accelerated transcoding (mejor rendimiento, especialmente con Plex Pass).	 Plex (más eficiente).
 Soporte y comunidad	Comunidad activa (foros, GitHub). Sin soporte oficial.	Soporte técnico prioritario para premium + comunidad grande.	 Empate (depende de necesidades).

PARTE PRÁCTICA

Guía Práctica: Instalación de Jellyfin en un Contenedor Docker (Alpine Linux)

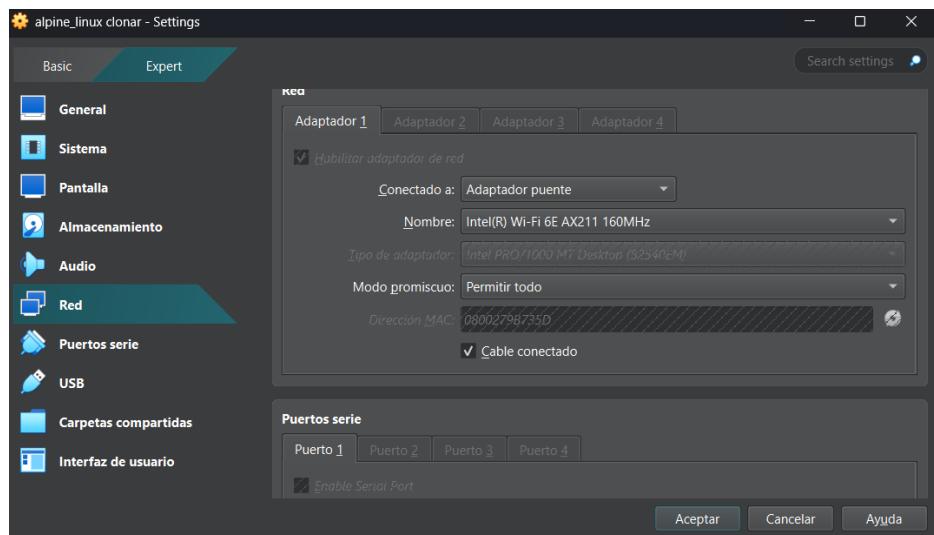
 **Requisitos Previos**

- Máquina virtual **Alpine Linux** configurada en VirtualBox (con Docker y Portainer instalados).
- **Dos adaptadores de red:**
 1. **RED NAT** (para acceso a internet).
 2. **Adaptador puente** (para conexión desde otros dispositivos, como tu móvil).

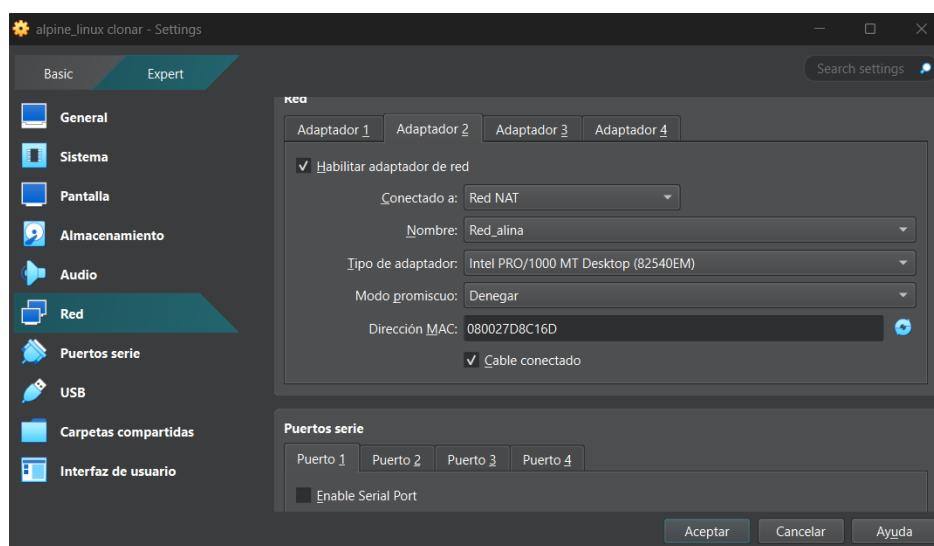
Paso 1: Configuración de Red en Alpine

1. Añade un segundo adaptador de red en VirtualBox:

- Ve a *Configuración > Red > Pestaña "Adaptador 1"*.
- Elige "**Adaptador puente**" y selecciona tu tarjeta de red inalámbrica.



RED NAT



- **Reinicia** la máquina virtual.
2. **Verifica las interfaces de red** en Alpine:

Usamos el comando de “ip a”

Deberías ver dos interfaces: eth0 (NAT) y eth1 (puente).

💡 Paso 2: Despliegue de Jellyfin con Docker Compose

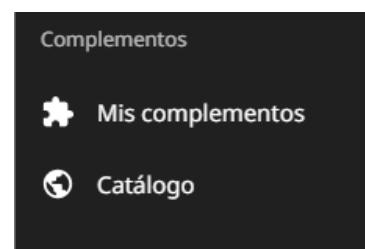
1. **Accede a Portainer** (<http://192.168.34.89:9443>).
2. Ve a **Stacks > Add Stack**.
3. **Copia y pega** el siguiente docker-compose.yml (fuente: punkymo.gitbook.io):

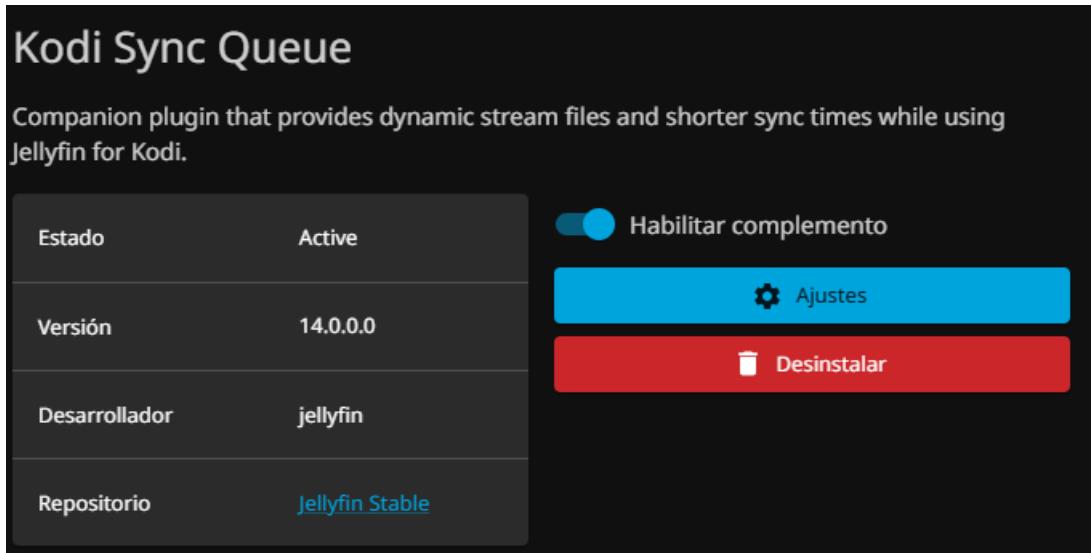
```
services:
  jellyfin:
    image: lscr.io/linuxserver/jellyfin:latest
    container_name: jellyfin
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=Etc/UTC
      - JELLYFIN_PublishedServerUrl=192.168.0.5 #optional
    volumes:
      - /path/to/library:/config
      - /path/to/tvseries:/data/tvshows
      - /path/to/movies:/data/movies
    ports:
      - 8096:8096
      - 8920:8920 #optional
      - 7359:7359/udp #optional
      - 1900:1900/udp #optional
    restart: unless-stopped
```

1. Haz clic en **Deploy the stack**.
2. Una vez creado, vemos que podemos acceder a nuestro contenedor Jellyfin mediante el puerto 8096.

💡 Paso 3: Instalación del Plugin "Kodi Sync Queue"

1. **Dirígete al Dashboard de Jellyfin**:
 - Haz clic en el ícono de **Configuración** (arriba a la derecha).
 - Selecciona "**Plugins**" en el menú lateral.





Descargar Contenido Multimedia

1. Descarga el archivo multimedia en mi equipo local:



Creamos un servidor Python con el comando: `python -m http.server` y luego

[wget http://192.168.34.64:8000/video-bea.mp4](http://192.168.34.64:8000/video-bea.mp4)

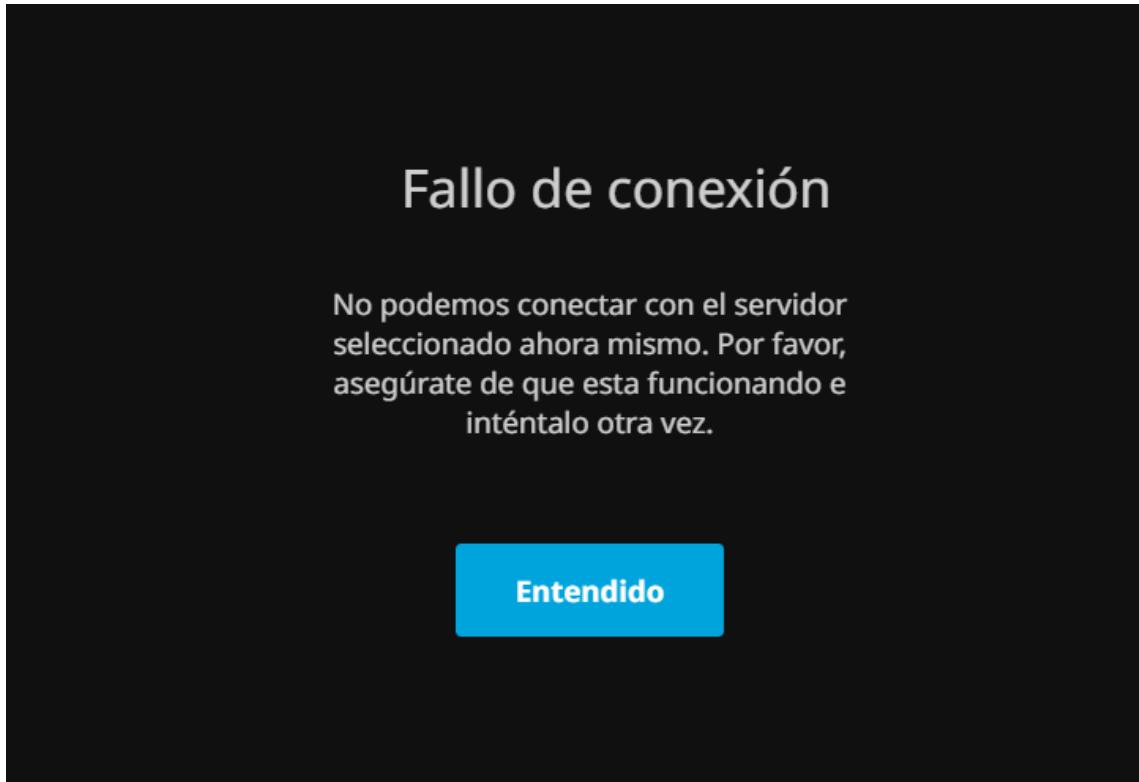
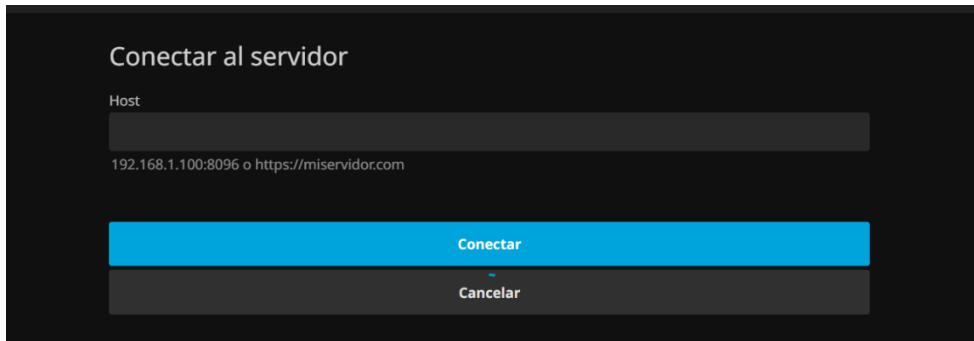
Paso 4: Mover el Archivo al Directorio de Jellyfin

- Copia el archivo** a la carpeta mapeada de Jellyfin (ej: /media/video/)

```
mv video-bea.mp4 /media/video/
```

INCIDENCIA

Nunca pude conectarme, lo intenté en casa lo repetí en clases, le pedí ayuda a Leonardo Duarte y no supimos que pudo suceder, Adriá Trillo



Paso 5: Actualizar la Biblioteca de Jellyfin

1. Escaneo automático:

- Jellyfin detectará el nuevo archivo en segundos (si el escaneo automático está activado).

2. Forzar escaneo manual:

- Ve a <http://192.168.34.89:8096> > **Bibliotecas** > **Escaneo de biblioteca**

Paso 6: Configura IP Forwarding (Conectar desde el Móvil)

Habilita el reenvío de paquetes en Alpine:

```
echo "net.ipv4.ip_forward=1" >> /etc/sysctl.conf  
sysctl -p
```

Configura iptables para permitir tráfico entre redes:

```
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE  
iptables -A FORWARD -i eth1 -o eth0 -j ACCEPT
```

Guarda las reglas (opcional, depende de tu setup):

```
apk add iptables-persistent  
service iptables save
```

Paso 7: Acceso desde el Móvil

1. **Conéctate a la misma red** que el adaptador puente (eth1).
2. Abre un navegador y accede a:

<http://192.168.34.89:8096>

Conclusión: Aprendizaje Valioso a Través de Desafíos

Aunque **no logramos conectar exitosamente al servidor Jellyfin**, este proceso fue una **experiencia de aprendizaje enriquecedora** que destacó la importancia de la resiliencia técnica y la solución de problemas. Aquí el resumen:

Incidencia Principal

- **Fallo de conexión al servidor:**

- Posibles causas: Configuración incorrecta de red en Alpine, reglas de iptables mal aplicadas, puertos bloqueados o errores en el despliegue del contenedor.
- *¿Qué intentamos?:* Verificamos adaptadores de red, IP Forwarding y accesibilidad vía wget/servidor Python.

Lecciones Aprendidas

1. **Redes en VirtualBox/Alpine:**

- La configuración de múltiples adaptadores (NAT + puente) es clave para acceso local/remoto.
- Aprendimos a diagnosticar conexiones con ip a, ping y netstat.

2. **Docker y Volúmenes:**

- Revisamos la importancia de mapear rutas correctamente en docker-compose.yml para evitar pérdida de datos.

3. **Herramientas de Transferencia:**

- Usamos alternativas como python -m http.server y wget para mover archivos entre sistemas.

4. **Soporte y Comunidad:**

- Documentamos cada paso para facilitar la depuración y futuras consultas en foros como [Jellyfin Discourse](#).

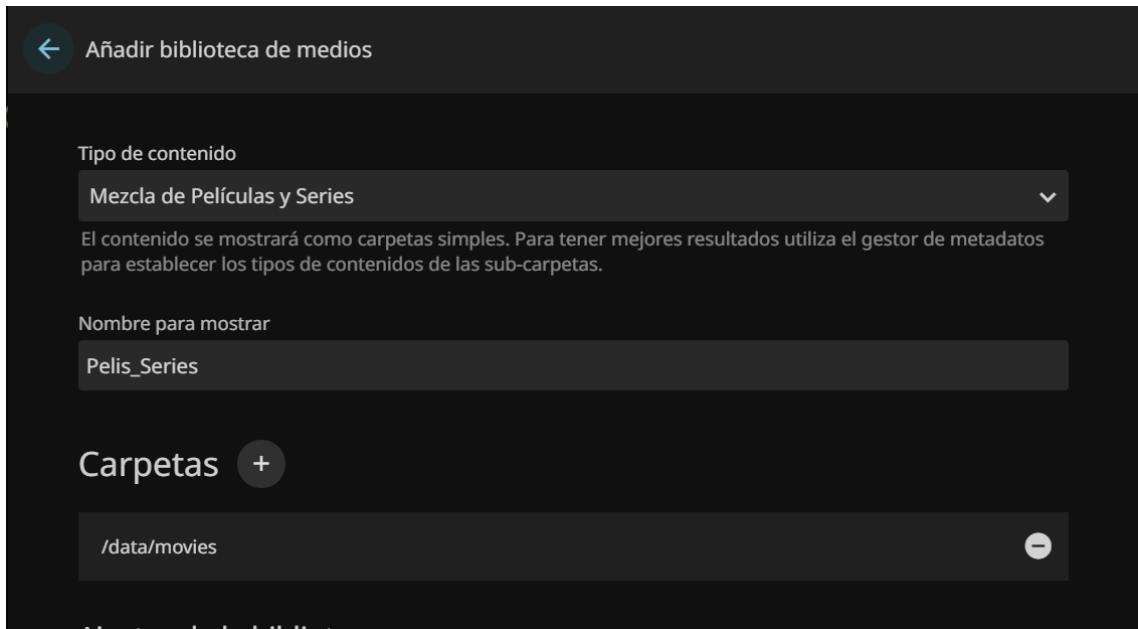
Jellyfin con Docker-Compose parte 2

ANEXOS:

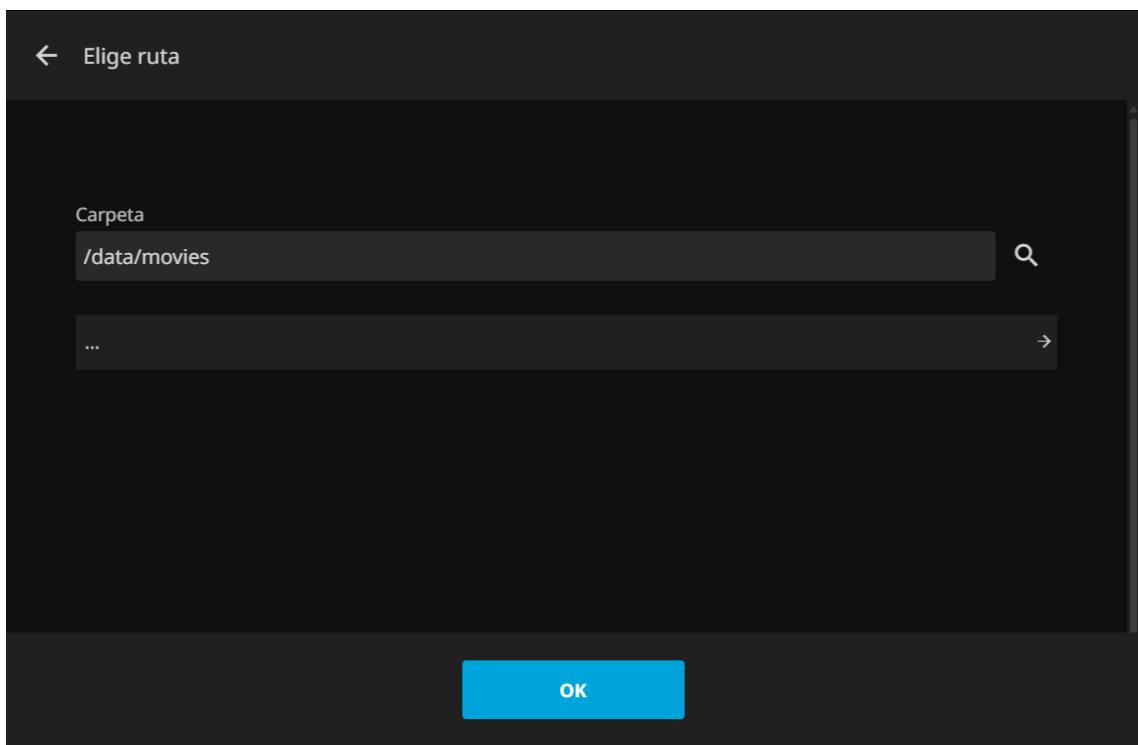
Nos pasamos de una maquina a otra el Docker compose

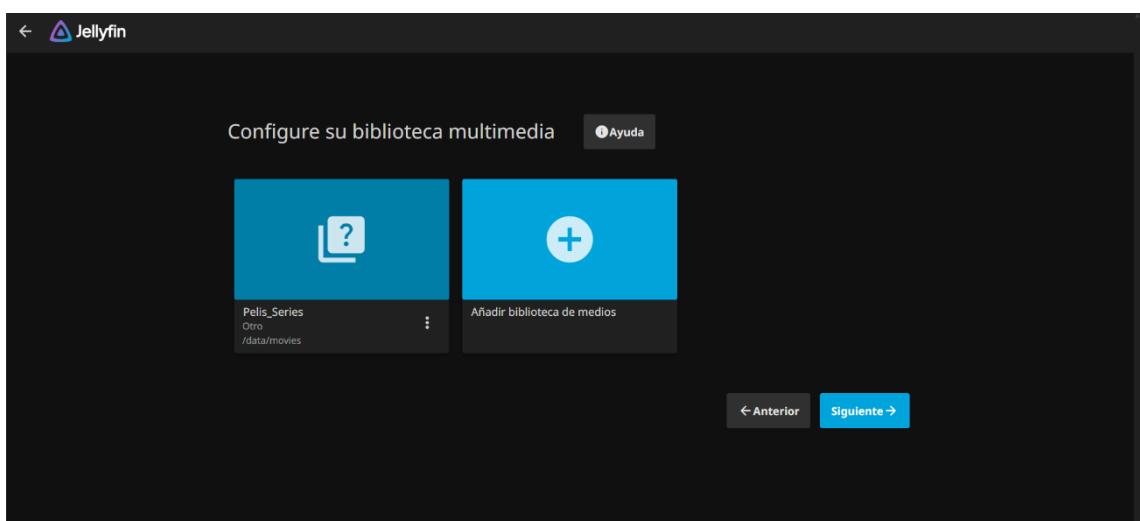
```
C:\Users\beasl>scp "C:\Users\beasl\Downloads\docker-compose.yml" root@192.168.1.71:/root/
root@192.168.1.71's password:
docker-compose.yml                                         100%   240     33.5KB/s   00:00
```

Configuramos la biblioteca de medios



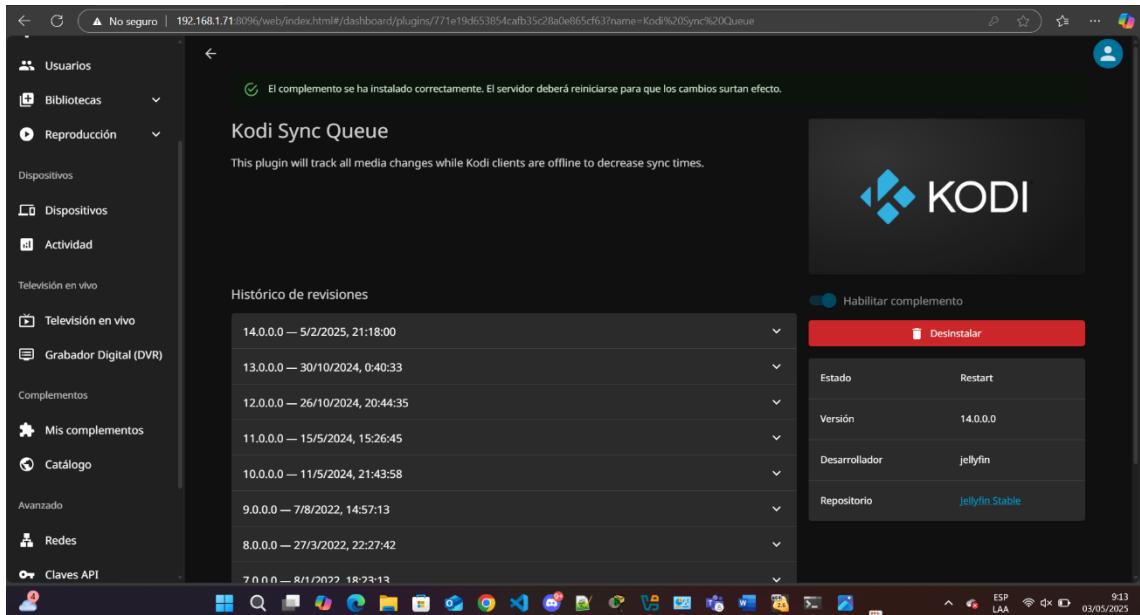
En la siguiente ruta





Descargamos el plugin de Kodi Sync Queue

A screenshot of the Jellyfin dashboard. On the left is a sidebar with links like Usuarios, Bibliotecas, Reproducción, Dispositivos, Actividad, Televisión en vivo, Grabador Digital (DVR), Complementos, Mis complementos, Catálogo, Avanzado, and Redes. The main content area shows the "Kodi Sync Queue" plugin page. It features a "Historial de revisiones" table with versions from 14.0.0.0 to 7.0.0.0. To the right is a summary card for the plugin, showing the KODI logo, a note that Jellyfin needs to be restarted after installation, and a large "Instalar" button. The card also displays the plugin's state as "No instalada", version "14.0.0.0", developer "jellyfin", and repository "Jellyfin Stable".



Nos pasamos el video

```
PS C:\Users\beasl> scp "C:\Users\beasl\Downloads\Bomba Estéreo, Rawayana, ASTROPICAL - Me Pasa (Piscis) (Official Video).mp4" root@192.168.1.71:/root/
root@192.168.1.71's password:
Bomba Estéreo, Rawayana, ASTROPICAL - Me Pasa (Piscis) (Official Video).mp4          100%  262MB   5.4MB/s   00:48
PS C:\Users\beasl>
```

Creamos los directorios

```
localhost:~# mkdir -p /data/movies/tvshows/
localhost:~# mv "/root/Bomba Estéreo, Rawayana, ASTROPICAL - Me Pasa (Piscis) (Official Video).mp4" /data/movies/tvshows/
localhost:~# ls -l /data/movies/tvshows/
total 267912
-rw-r--r--  1 root      root      274341188 May  3 09:19 Bomba Estéreo, Rawayana, A
STROPICAL - Me Pasa (Piscis) (Official Video).mp4
localhost:~#
```

Verificamos que se pasara correctamente

```
localhost:~# ls -la
total 267932
drwx-----  3 root      root      4096 May  3 09:18 .
drwxr-xr-x  22 root      root      4096 Apr 25 17:54 ..
-rw-----  1 root      root      1580 May  3 09:19 .ash_history
drwx-----  2 root      root      4096 Dec 18 15:59 .ssh
-rw-r--r--  1 root      root      274341188 May  3 09:19 Bomba Estéreo, Rawayana, ASTROPICAL - Me Pasa (Piscis) (Officia
l Video).mp4
-rw-r--r--  1 root      root      240 May  3 08:56 docker-compose.yml
localhost:~#
```

Visualizamos en la web el video.

