

Installation and configuration guide

- [Setup \(install/update\)](#)
- [License Activation](#)
- [Setup guide: Jellyfin setup](#)
- [Add server \(Jellyfin server\)](#)
- [Service configuration](#)
- [Add new library/item in Jellyfin](#)
- [Getting Access Hash \(API key\)](#)

Setup (install/update)

Jellyfin module **WISECP**

[Order now](#) | [Download](#) | [FAQ](#)

To install and update a module, you must perform one and the same action.

1. Download the latest version of the module.

```
wget http://download.puqcloud.com/WISECP/Product/PUQ_WISECP-Jellyfin/PUQ_WISECP-Jellyfin-latest.zip
```

All versions are available: https://download.puqcloud.com/WISECP/Product/PUQ_WISECP-Jellyfin/

2. Unzip the archive with the module.

```
unzip PUQ_WISECP-Jellyfin-latest.zip
```

3. Copy and Replace "puqJellyfin" from "PUQ_WISECP-Jellyfin" to "WISECP_WEB_DIR/coremio/modules/Product/"

License Activation

Jellyfin module **WISECP**

[Order now](#) | [Download](#) | [FAQ](#)

1. Log in to the administrative area of your **WISECP**.
2. Go to module configuration.

Services -> Service Management -> Module Settings -> Other -> All Modules -> PUQ Jellyfin





Other

Initial Settings

All Modules

Search Within Modules

Write something...



PUQ Jellyfin



PUQ Jellyfin

License key

10

AF



2024-01-01T13:06:09+01:00



Check and Save

Version: 1.0



Web Site



Documentation



Changelog

+ Add New Server

Server Groups

Setup guide: Jellyfin setup

Jellyfin module **WISECP**

[Order now](#) | [Download](#) | [FAQ](#)

1. Install Jellyfin on your server

To simplify deployment and help automate this for as many users as possible, we provide a BASH script to handle repo installation as well as installing Jellyfin. All you need to do is run this command on your system

```
wget -O- https://repo.jellyfin.org/install-debuntu.sh | sudo bash
```

2. SSL certificate generation:

```
sudo apt install nginx certbot python3-certbot-nginx
```

```
sudo certbot --nginx --agree-tos --redirect --hsts --staple-ocsp --email YOUR_EMAIL -d  
DOMAIN_NAME
```

```
echo "0 0 * * * root certbot renew --quiet --no-self-upgrade --post-hook 'systemctl reload  
nginx'" | sudo tee -a /etc/cron.d/renew_certbot
```

3. Configure nginx from a subdomain

Create a file named jellyfin.conf.

```
cd /etc/nginx/conf.d/  
nano jellyfin.conf
```

Then, insert the following text, replacing **DOMAIN_NAME** with your domain.

```

# Uncomment the commented sections after you have acquired a SSL Certificate
server {
    listen 80;
    listen [::]:80;
    server_name DOMAIN_NAME;

    # Uncomment to redirect HTTP to HTTPS
    return 301 https://$host$request_uri;
}

server {
    listen 443 ssl http2;
    listen [::]:443 ssl http2;
    server_name DOMAIN_NAME;

    ## The default `client_max_body_size` is 1M, this might not be enough for some posters, etc.
    client_max_body_size 20M;

    # use a variable to store the upstream proxy
    # in this example we are using a hostname which is resolved via DNS
    # (if you aren't using DNS remove the resolver line and change the variable to point to an IP
    address e.g `set $jellyfin 127.0.0.1`)
    set $jellyfin 127.0.0.1;
    resolver 127.0.0.1 valid=30;

    ssl_certificate /etc/letsencrypt/live/DOMAIN_NAME/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/DOMAIN_NAME/privkey.pem;
    include /etc/letsencrypt/options-ssl-nginx.conf;
    ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem;
    add_header Strict-Transport-Security "max-age=31536000" always;
    ssl_trusted_certificate /etc/letsencrypt/live/DOMAIN_NAME/chain.pem;
    ssl_stapling on;
    ssl_stapling_verify on;

    # Security / XSS Mitigation Headers
    # NOTE: X-Frame-Options may cause issues with the webOS app
    add_header X-Frame-Options "SAMEORIGIN";
    add_header X-XSS-Protection "0"; # Do NOT enable. This is obsolete/dangerous
    add_header X-Content-Type-Options "nosniff";

```

```

## COOP/COEP. Disable if you use external plugins/images/assets
add_header Cross-Origin-Opener-Policy "same-origin" always;
add_header Cross-Origin-Embedder-Policy "require-corp" always;
add_header Cross-Origin-Resource-Policy "same-origin" always;

## Permissions policy. May cause issues on some clients
add_header Permissions-Policy "accelerometer=(), ambient-light-sensor=(), battery=(),
bluetooth=(), camera=(), clipboard-read=(), display-capture=(), document-domain=(), encrypted-
media=(), gamepad=(), geolocation=(), gyroscope=(), hid=(), idle-detection=(), interest-
cohort=(), keyboard-map=(), local-fonts=(), magnetometer=(), microphone=(), payment=(),
publickey-credentials-get=(), serial=(), sync-xhr=(), usb=(), xr-spatial-tracking=()" always;

## Tell browsers to use per-origin process isolation
add_header Origin-Agent-Cluster "?1" always;

## Content Security Policy
## See: https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP
## Enforces https content and restricts JS/CSS to origin
## External Javascript (such as cast_sender.js for Chromecast) must be whitelisted.
## NOTE: The default CSP headers may cause issues with the webOS app
add_header Content-Security-Policy "default-src https: data: blob: http://image.tmdb.org;
style-src 'self' 'unsafe-inline'; script-src 'self' 'unsafe-inline' https://www.gstatic.com
https://www.youtube.com blob:; worker-src 'self' blob:; connect-src 'self'; object-src
'none'; frame-ancestors 'self'";

location = / {
    return 302 http://$host/web/;
    ##return 302 https://$host/web/;
}

location / {
    ## Proxy main Jellyfin traffic
    proxy_pass http://$jellyfin:8096;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
    proxy_set_header X-Forwarded-Protocol $scheme;
    proxy_set_header X-Forwarded-Host $http_host;

```

```

    # Disable buffering when the nginx proxy gets very resource heavy upon streaming
    proxy_buffering off;
}

# location block for /web - This is purely for aesthetics so /web/# / works instead of having
to go to /web/index.html/# /
location = /web/ {
    # Proxy main Jellyfin traffic
    proxy_pass http://$jellyfin:8096/web/index.html;
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
    proxy_set_header X-Forwarded-Protocol $scheme;
    proxy_set_header X-Forwarded-Host $http_host;
}

location /socket {
    # Proxy Jellyfin Websockets traffic
    proxy_pass http://$jellyfin:8096;
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection "upgrade";
    proxy_set_header Host $host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;
    proxy_set_header X-Forwarded-Protocol $scheme;
    proxy_set_header X-Forwarded-Host $http_host;
}
}

```

Restarting nginx

```
sudo service nginx restart
```

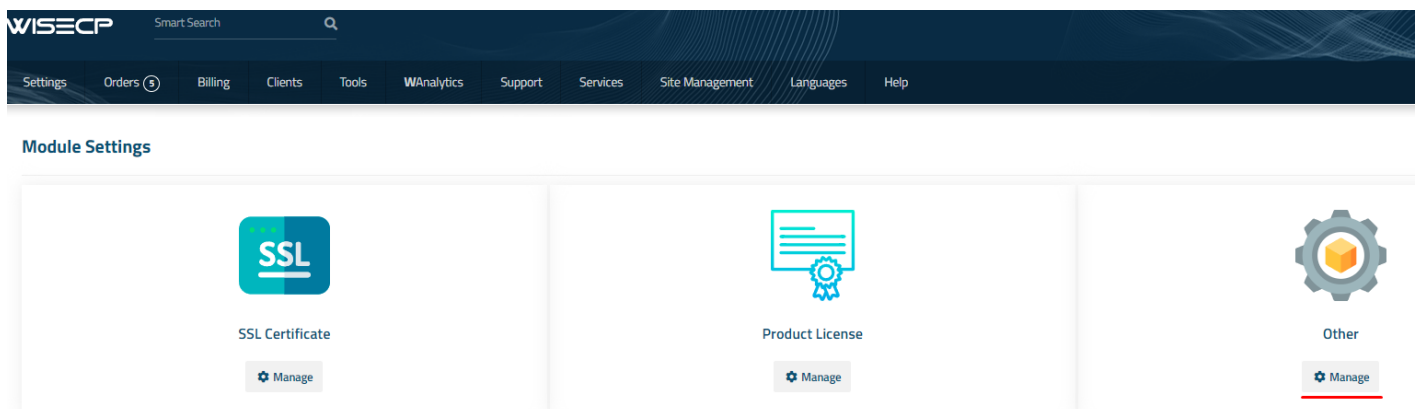
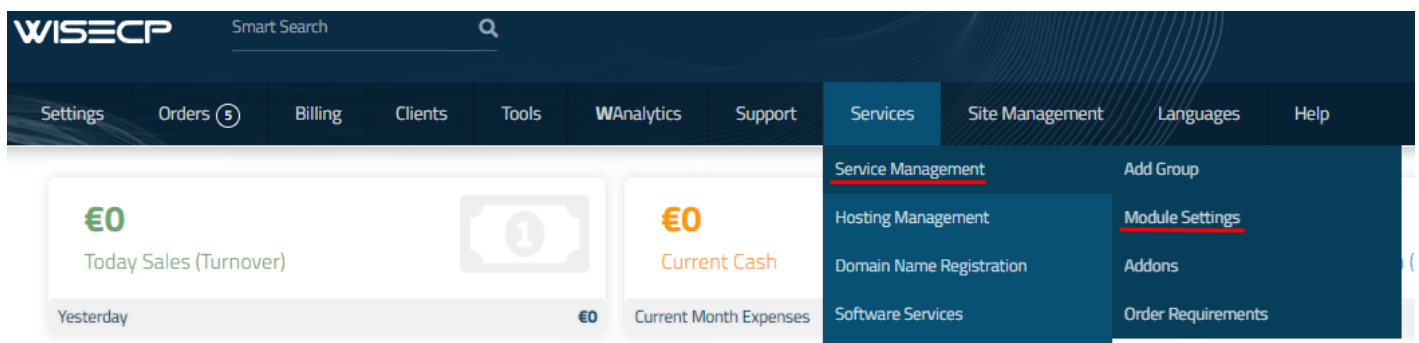

Add server (Jellyfin server)

Jellyfin module **WISECP**

[Order now](#) | [Download](#) | [FAQ](#)

1. Log in to the administrative area of your **WISECP**.
2. Go to module configuration.

Services -> Service Management -> Module Settings -> Other -> All Modules -> PUQ Jellyfin



Other

Initial Settings

All Modules

Search Within Modules

Write something...

PUQ Jellyfin

3. In the opened page, click the '**Add Server**' button.

Initial Settings

All Modules

PUQ Jellyfin

License key

1QAASZ-ZBCXZX-FL8H9A-WKVVQA-UAHZAF

2024-01-01T13:06:09+01:00

Check and Save

Version: 1.0

Web Site

Documentation

Changelog

Add New Server

Server Groups

Servers Jellyfin

10

Show Record

Search

#	Name	Status	Group	Default	IP/Domain	Usage Stats	
6	jellyfin-test.uuq.pl	active	Default		jellyfin-test.uuq.pl	2/111	<div></div> <div></div>

Showing records from 1 to 1 - 1

Prev

1

Next


Turn Back

4. On the opened page, enter all the necessary information:

- **Name:** Displayed name of the server.
- **Maximum Number of Accounts:** The number of services that can be on this server.
- **Server Group:** Optionally, choose the server group.
- **IP Address or Domain:** The address of the Emby server you are connecting to.

- **Username:** is the username for the account on [Jellyfin](#).
- **Password:** is the password for the account on [Jellyfin](#).
- **Access Hash:** API key that you created in the previous step on the Emby server.
- Check the **SSL** box if you want to use SSL-encrypted connection. If necessary, specify the port and perform a connection test.

Initial Settings
All Modules



PUQ Jellyfin

< Go to Back

Add New Server

Name
jellyfin-test.uuq.pl

Maximum Number of Accounts
100

Server Group
Default

Default
☐ Set as default server in group

IP Address or Domain
jellyfin-test.uuq.pl

Username
admin

Password

Access Hash

SSL
☒ Connect using SSL

Port
443
☐ Change standard port

Test Connection
Test Connection

Connection test Successful

Add New Server

Service configuration

Jellyfin module **WISECP**

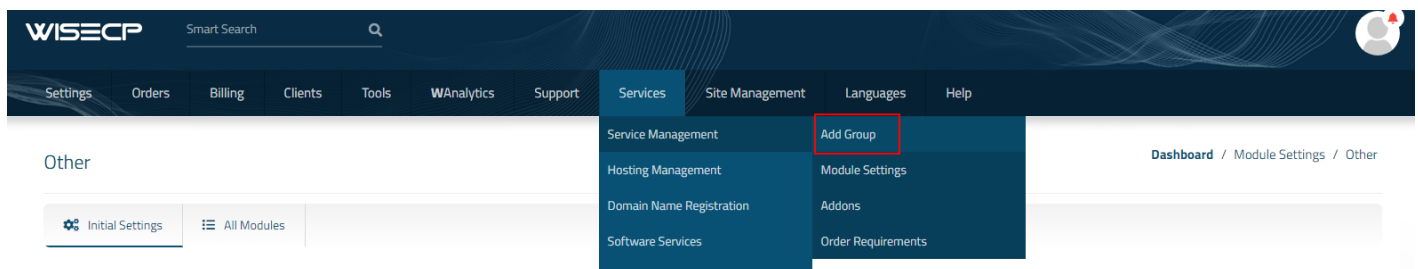
[Order now](#) | [Download](#) | [FAQ](#)

If you do not have a **Service Group** where you want to place the new service, you need to create a new **Service Group**

1. Log in to the administrative area of your **WISECP**.
2. Create New Service Group

Go to

Services -> Service Management -> Add Group



Enter all the necessary data and click the **'Create Group'** button.

WISECP

Smart Search

SettingsOrdersBillingClientsToolsWAnalyticsSupportServicesSite ManagementLanguagesHelp

Create New Service Group

Dashboard / Create New Service Group

Create a new service group, you can also create new categories and manage the existing one from here.

ENUKRURDPTPLNLLVKAITIDHUFRAESLEDECSBSAR

Title

Media

Short Detail

Short Group Description

Listing Template

☒ Box List ☐ Horizontal List

Status

Enable

Background Image

(optional)

Group Colour

#FFFFFF

Order Upgrades

☐ Allow clients to upgrade to higher package in the same service group

External HTML Code

Paste your HTML code to this field (Optional)

You can optionally add external HTML code and customize it. There should not be any incorrect or incomplete HTML codes, otherwise page structure will be empty

3. Adding a New Service

Go to

Services -> our service group where you need to add the new service.

WISECP

Smart Search

SettingsOrdersBillingClientsToolsWAnalyticsSupportServicesSite ManagementLanguagesHelp

€0

Today Sales (Turnover)

Yesterday

€0

€0

Current Cash

Current Month Expenses

€0

0

Overdue Invoice (Total €0)

Payment Date Upcoming (Total €0 / Last 7 Days)

0

Client Registration (Today)

Yesterday

0

0

Pending Support Tickets

Total Resolved

0

0

Order in the Process (Total €0)

0

Reminder (Will Happen Today)

Service Management

Hosting Management

Domain Name Registration

Software Services

International SMS Services

Google SEO Services

VPN

Media

In the opened window, click the **'Create New Service'** button.

On the opened page, enter all the necessary details for your new service and navigate to the 'Core' tab.

Select the 'PUQ Jellyfin' module from the drop-down list of modules.

4. Fill in the configuration options according to your preferences.

Libraries:

- **Use All Libraries:** Choosing all of libraries from your Jellyfin server
- **Libraries:** The libraries you want to make available to customers of this product.

(IMPORTANT! Start a new line for each new library)

Example:

"Movies

Beginner`s Programming Course

Comedy"

IMPORTANT!

If you have a folder named "-", please enter it not as the first one or rename it.
If you don't want any folder to be accessible, type "-".

User Configuration:

- **Streaming bitrate limit:** An optional per-stream bitrate limit for all out of network devices. This is useful to prevent devices from requesting a higher bitrate than your internet connection can handle. This may result in increased CPU load on your server in order to transcode videos on the fly to a lower bitrate.
- **SyncPlay access:** The SyncPlay feature enables to sync playback with other devices. Select the level of access this user has to the SyncPlay
- **Remote control:** Remote control of shared devices (DLNA devices are considered shared until a user begins controlling them)
- **Media downloads:** Users can download media and store it on their devices. This is not the same as a sync feature. Book libraries require this enabled to function properly.
- **User sessions:** Set the maximum number of simultaneous user sessions.
A value of 0 will disable the feature.
- **Failed login attempts:** Determine how many incorrect login tries can be made before lockout occurs.
A value of zero means inheriting the default of three tries for normal users and five for administrators. Setting this to -1 will disable the feature.
- **Username prefix/Username suffix:** Necessary in order to generate a username for the service, in the format: **prefix<client_id>-<service_id>suffix**

Allow playback (Restricting access to transcoding may cause playback failures in clients due to unsupported media formats)

- **media**
- **audio that requires transcoding**
- **video that requires transcoding**
- **video that requires conversion without re-encoding**

Feature access:

- **Allow Live TV access**
- **Allow Live TV recording management**
- **Force transcoding of remote media sources such as Live TV**

Links

- **Link to instruction:** Link to the instruction, if filled out, it will be reflected in the client area
-

Add new library/item in Jellyfin

Jellyfin module **WISECP**

[Order now](#) | [Download](#) | [FAQ](#)

New library:

To create a new library, you should to **create a folder** in your jellyfin server.

First, navigate to the root of your project "/" and enter the following commands:

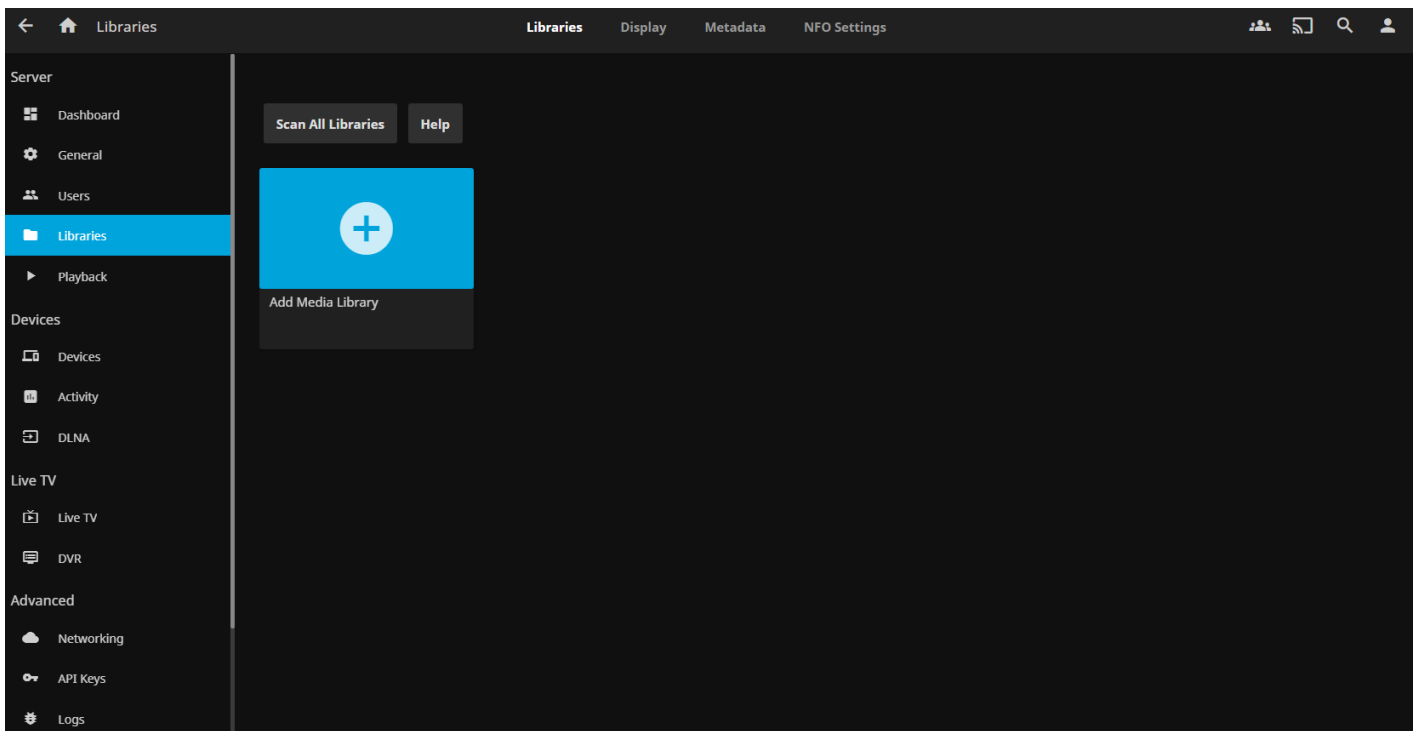
```
cd mnt
```

```
mkdir LIBRARY_NAME
```

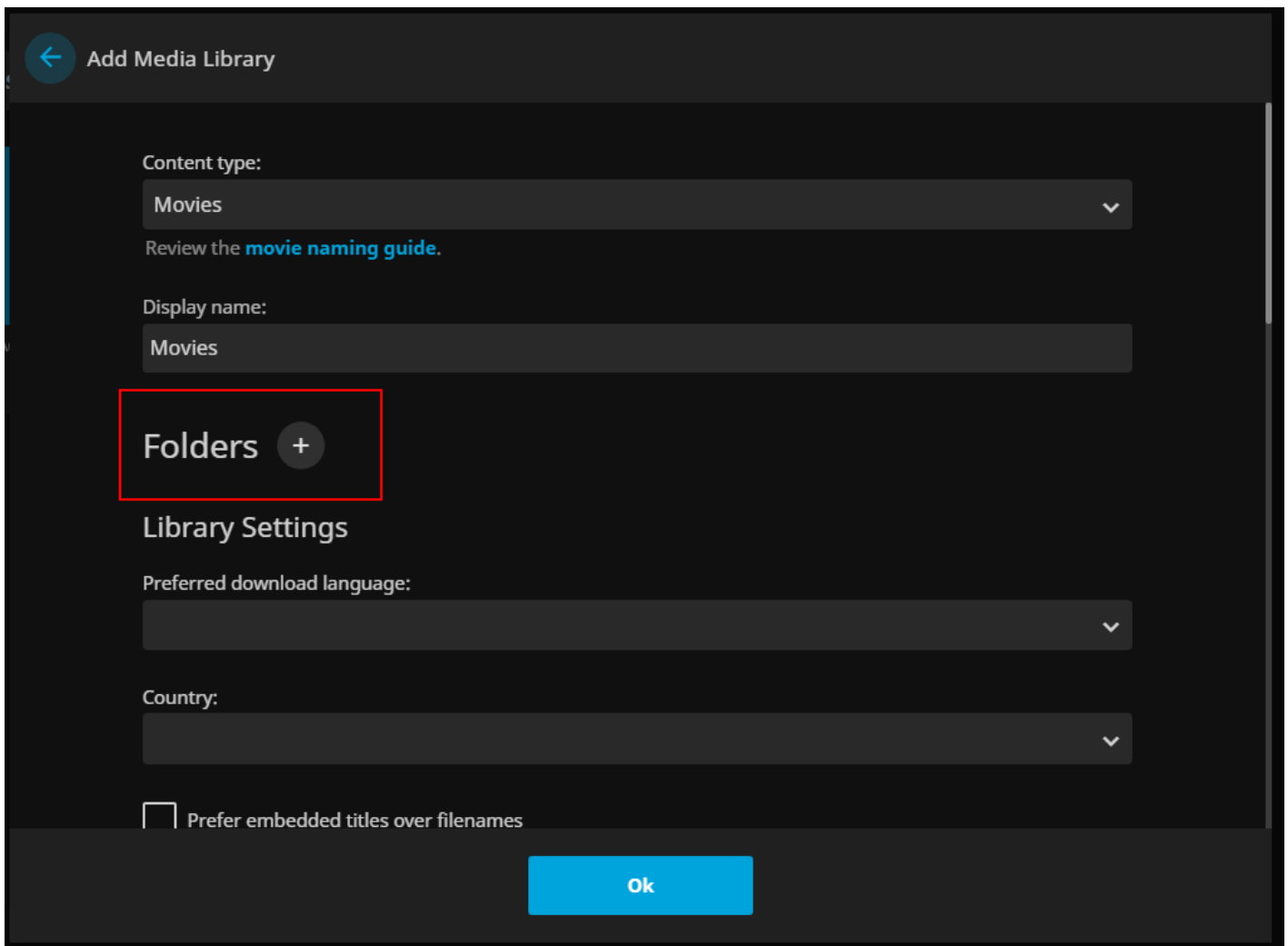
LIBRARY_NAME - The name of the new library.

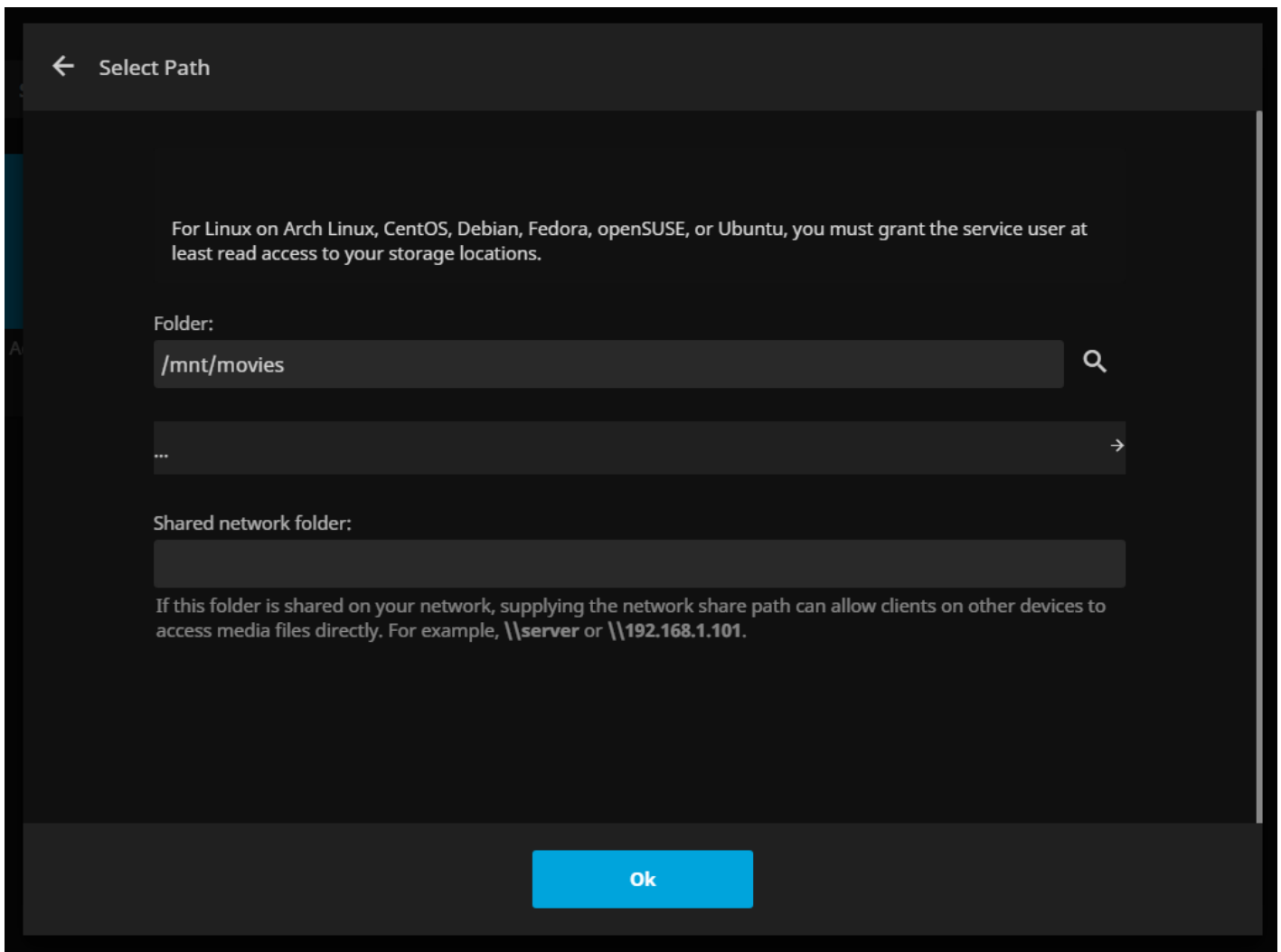
After creating a folder, let's create a library in Jellyfin:

Add a new library:



Assign the appropriate folder to this library:





New item:

To download a new item, navigate to the root of your project "/" and enter the following commands:

```
cd mnt/LIBRARY_NAME
```

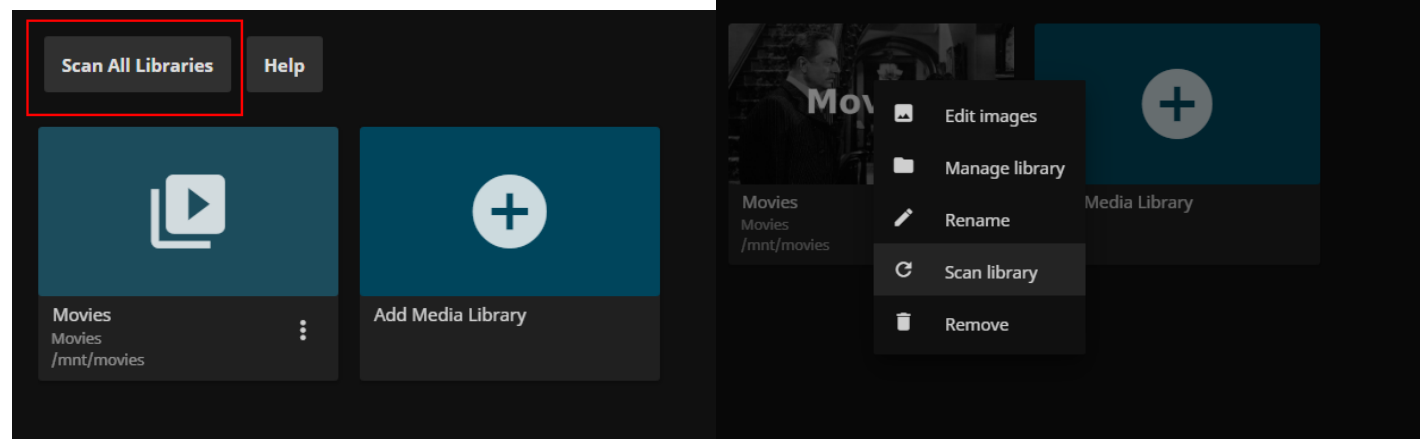
```
wget DOWNLOAD_ITEM_URL
```

LIBRARY_NAME - The name of the library.

DOWNLOAD_ITEM_URL - The download URL of the item you want to upload to the server.

After all

Scan all libraries by clicking on the "Scan All Libraries" button. To scan a specific library, click on the three dots and select "Scan Library".



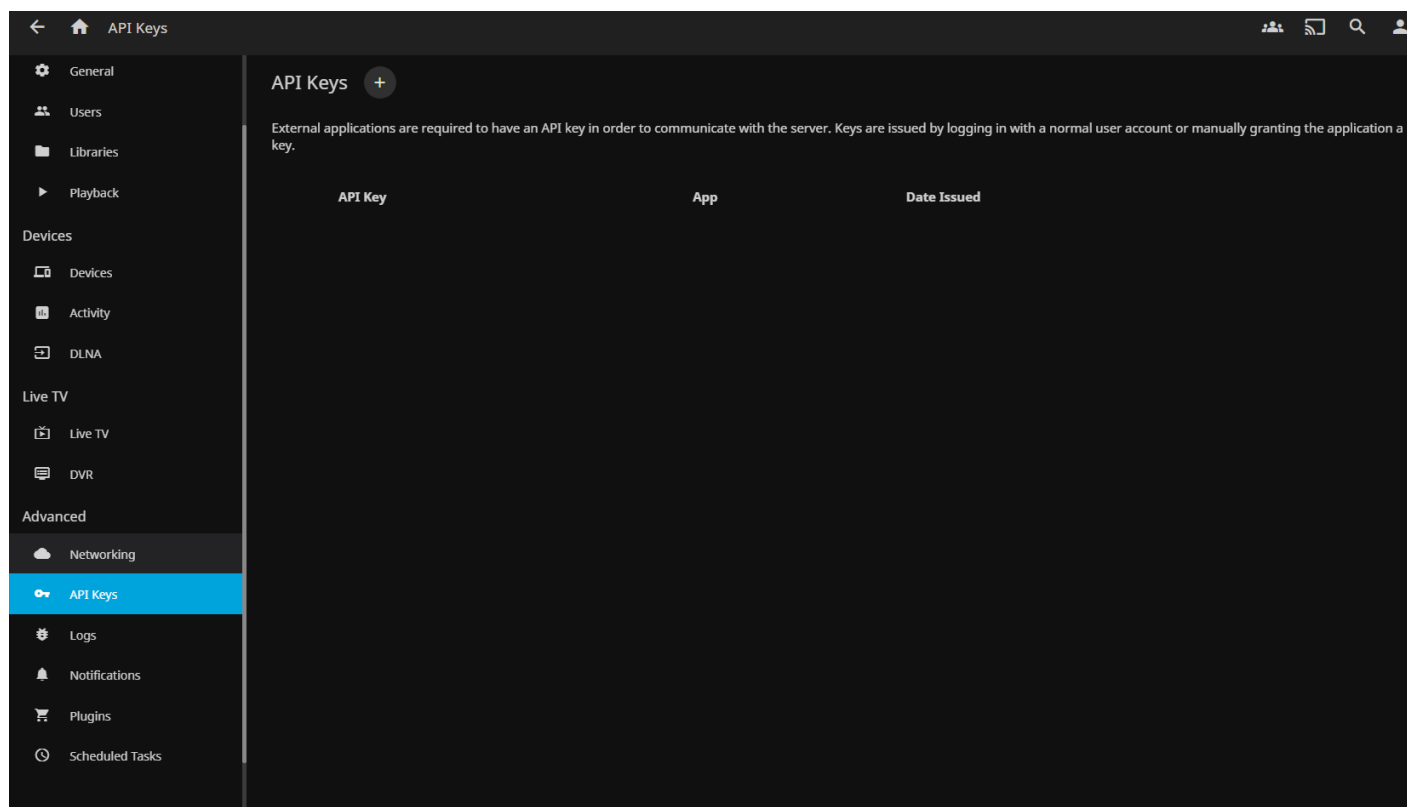
Getting Access Hash (API key)

Jellyfin module **WISECP**

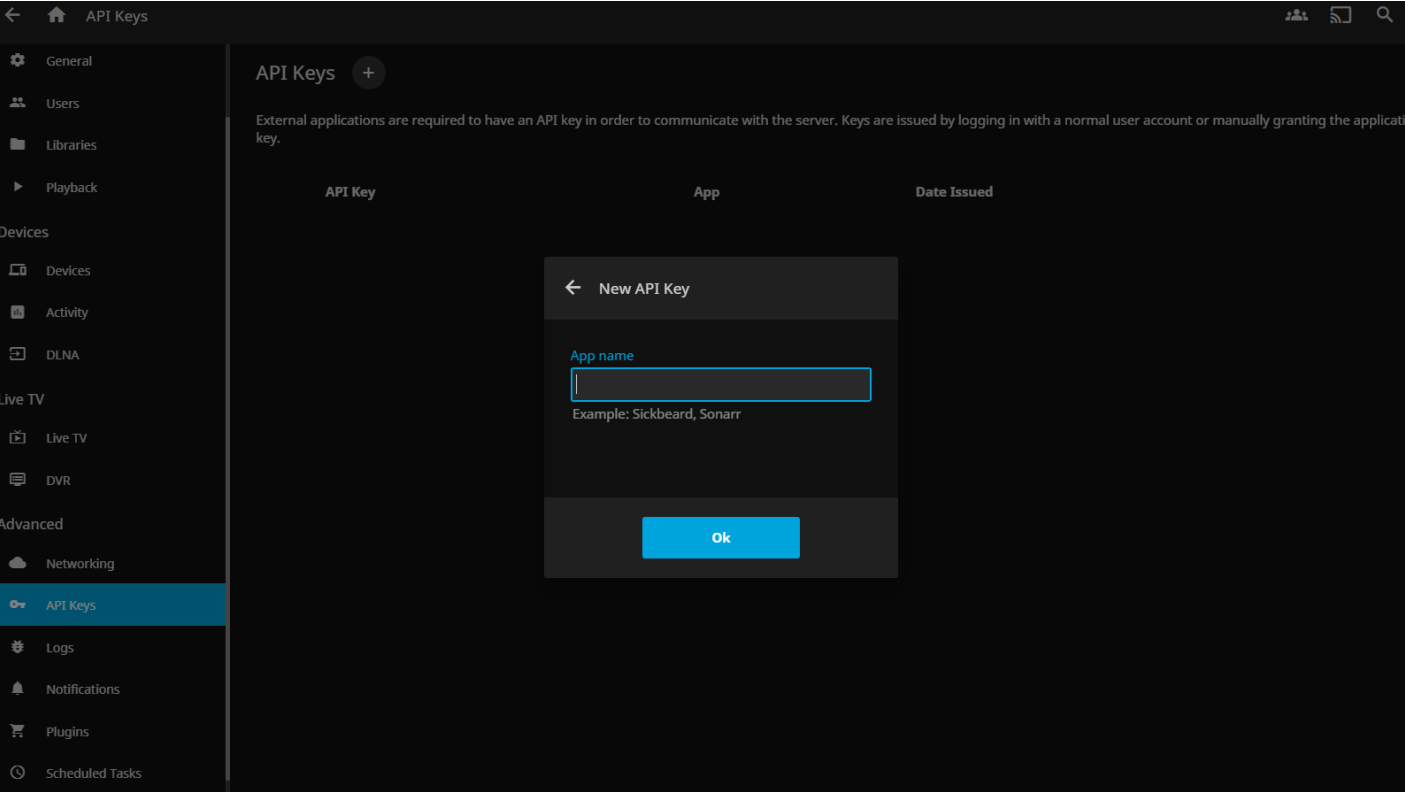
[Order now](#) | [Download](#) | [FAQ](#)

Dashboard -> API Keys

Create a new API Key



Whrite an App Name



Now you can getting an API key

