

# Ombi

is a self-hosted web application that automatically gives your shared Plex or Emby users the ability to request content by themselves! Features include:

- Customisable content requesting system
- Allows your users to receive custom notifications
- Real-time updating. Will show if the request is already on Plex or even if it's already monitored.
- Slick, responsive and mobile-friendly UI

For more information about this app, you may visit the app's [GitHub repo](#).

In this setup, we'll be setting up Ombi for the first time. This assumes the following:

- You've set up a content manager. Ombi supports Sonarr, Radarr, CouchPotato and SickChill.
- You have a media server installed. Ombi supports Plex, Emby and Jellyfin.
- Your username for Ombi will be your service username and the password you set during the install procedure.

## Setting up Ombi

- Click the `connect` button to connect to your Ombi WebUI. Type your username and the password you created upon installation.
- It will be good practice here to set your `User Management` defaults. You can do this by going to `Settings` -> `Configuration` -> `User Management` and select your **default roles**.
- Select **User Management** from the main pane on the left side to add users. These users will be able to access your Ombi url at `https://{user}.{server}.usbx.me/ombi` to request content.
- Ombi's only other setup really just consists of connecting all of your applications that you have already set up before-hand. You will need only the API keys for your indexers on hand. Please see the tables below for the various application examples.

## Connection Details

### Media Server

#### Plex

- Enter Plex `Username` and `Password`, then click `Load Servers`.
- Select your Plex server from the dropdown.
- If asked for Plex auth token, see [this guide](#) for how to find it.
- Under Plex Server Configuration, change `Hostname or IP` to `172.17.0.1` if both Ombi and Plex Media Server are on the same Ultra.cc Service.
- Select Load Libraries and select the libraries you want

#### Emby/Jellyfin

Emby

Hostname or IP: {username}.{servername}.usbx.me  
Port: 443  
SSL: Checked  
Emby API Key: {Emby API Key}  
Externally Facing Hostname: https://{username}.{servername}.usbx.me/emby

#### Jellyfin

Hostname or IP: {username}.{servername}.usbx.me  
Port: 443  
SSL: Checked  
Emby API Key: {Jellyfin API Key}  
Base URL: /jellyfin  
Externally Facing Hostname: https://{username}.{servername}.usbx.me/jellyfin

## Sonarr

Enable: Checked  
V3: Checked  
Hostname or IP: {username}.{servername}.usbx.me  
Port: 443  
API Key: {Sonarr API Key}  
SSL: Checked  
Base URL: /sonarr

## Radarr

Enable: Checked  
Hostname or IP: {username}.{servername}.usbx.me  
Port: 443  
API Key: {Radarr API Key}  
SSL: Checked  
Base URL: /radarr

## CouchPotato SSL

Enable: Checked  
Hostname or IP: {username}.{servername}.usbx.me  
Port: 443  
API Key: {CouchPotato API Key}  
SSL: Checked  
Base URL: /{username}/couchpotato

# Troubleshooting Information

## Why are Ombi's search function not working?

Recent versions of Ombi, when upgraded, had some issues with their database. To fix this, run the following command on your slot's shell.

```
rm -f "$HOME"/.apps/ombi/OmbiExternal.db && app-ombi upgrade
```

## Why are Ombi's requests to Radarr and/or Sonarr not working?

Please check your Ombi connection details to make sure they are correct. You can find the correct connection details in the previous section called [Connection Details](#) and compare it to yours.

---

Revision #25

Created 17 October 2019 09:32:24

Updated 6 May 2024 15:41:53 by varg