jus Public

Notifications Fork 2 Star 16

master Go to file Code

wantaekchoi and seeschloss Update PHP to 8.5.12 for CVE-... e0e2a93 · last month

cfg	Configurable contact ...	2 months ago
inc	PHP 8.5 compatibility	2 months ago
www	Configurable contact ...	2 months ago
Dockerfile	Update PHP to 8.5.12 ...	last month
LICENSE	GPLv3	10 years ago
README.md	Configurable contact ...	2 months ago

README GPL-3.0 license

jus - Just Upload Stuff

jus is a file upload/sharing system made to be effective and simple both for its users and its admins, without requiring Javascript, databases, or relying on redirects or extra unnecessary hoops.

Users can just upload one or multiple files, get a direct link to them and share it.

Admins can just put the source wherever they want, configure the two or three options, and clean up the old files any way they like.

Installation

Source and configuration should ideally live outside your document root, but if you don't know what this means just put everything in your web directory.

You can then rename the `www` directory to whatever you like, as well as the `index.php` file provided they stay at the same place relative to `inc/` and `cfg/`.

Copy `cfg/config.inc.php.dist` to `cfg/config.inc.php` and tweak it (more on this later).

PUT

If you want to be able to use PUT queries to upload files with `curl`, you will have to set `put.php` as the PUT handler script for Apache (it can work with Nginx as well, but you will have to research it yourself). This means enabling `mod_actions` and adding `Script PUT /put.php` in your virtual host section.

Free.fr

If you're going to host this on Free.fr's free hosting, you should first put the following in a file named `.htaccess` in your web directory:

```
<IfDefine Free>
php56 1
</IfDefine>
```

Docker

```
docker build . -t jus
docker run \
    -p 8080:80 \
    -e UPLOAD_SERVER=http://localhost:8080/files
    -e BASE_SERVER=localhost:8080 \
    -e CONTACT_EMAIL=contact@example.com \
    --mount type=volume,src=jus_files,dst=/var/w
    jus
```

Configuration

Then you should probably change the configuration defaults:

`upload_directory` is where the uploaded files with be saved. Use `__DIR__` to be sure to have a path relative to this config file. Ideally, this should be outside of the document root for jus.

`upload_server` is the prefix that will be given to uploaded files. If you have a different virtual host for serving uploaded files, which I recommend in order to make sure nobody messes up with your jus by uploading Javascript or HTML which will get executed in the browser security context of the main site, then this option should have the name of this file serving virtual host. In my case, jus lives at `up.ÿ.fr` and my `upload_server` is `down.ÿ.fr`, configured to have `upload_directory` as its document root.

`max_size` is the maximum size to accept for files. Don't forget to adjust your `php.ini` settings as well, relevant options are `post_max_size`, `upload_max_filesize` and `max_file_uploads`.

Usage

There are three different ways for a user to upload a file on jus.

The most obvious is clicking the *Select your file* button, selecting one or more files, and clicking *Up*. Once the upload is finished, the links to the files are shown.

If `PUT` is correctly configured on your server, another way is to type this from command-line:

About

jus - Just Upload Stuff

up.ÿ.fr

Readme

GPL-3.0 license

Activity

16 stars

3 watching

2 forks

Report repository

Releases

No releases published

Contributors 3

seeschloss SeeSchloß

n-e Nicolas

wantaekchoi wantaek

Languages

PHP 98.1% Dockerfile 1.9%

```
curl -T <file> https://url-to-jus
```

If it is not, curl can still upload easily enough using `POST`:

```
curl -F "file[]"=@<file> https://url-to-jus
```

When using `curl`, the output will just contain the paths to the upload files, one by line.

Administration

jus will put files in `upload_directory` under subdirectories, one for each day. So something uploaded on the 18th of August in 2017 would be in `files/2017-08-18`. If you want to limit the duration the files are available on your server, it should be easy enough to have a cron that every day removes the folder of today minus 30 days, or minus 365 or 12 000 days.

Something like:

```
rm -rf "/path/to/jus/files/$(date --date='30 days ago' +'%Y-%m-%d')"
```

in your cron.

Terms Privacy Security Status Community Docs Contact Manage cookies
Do not share my personal information
© 2026 GitHub, Inc.