FileSender - API (filesender branch v3)

1. Overall information

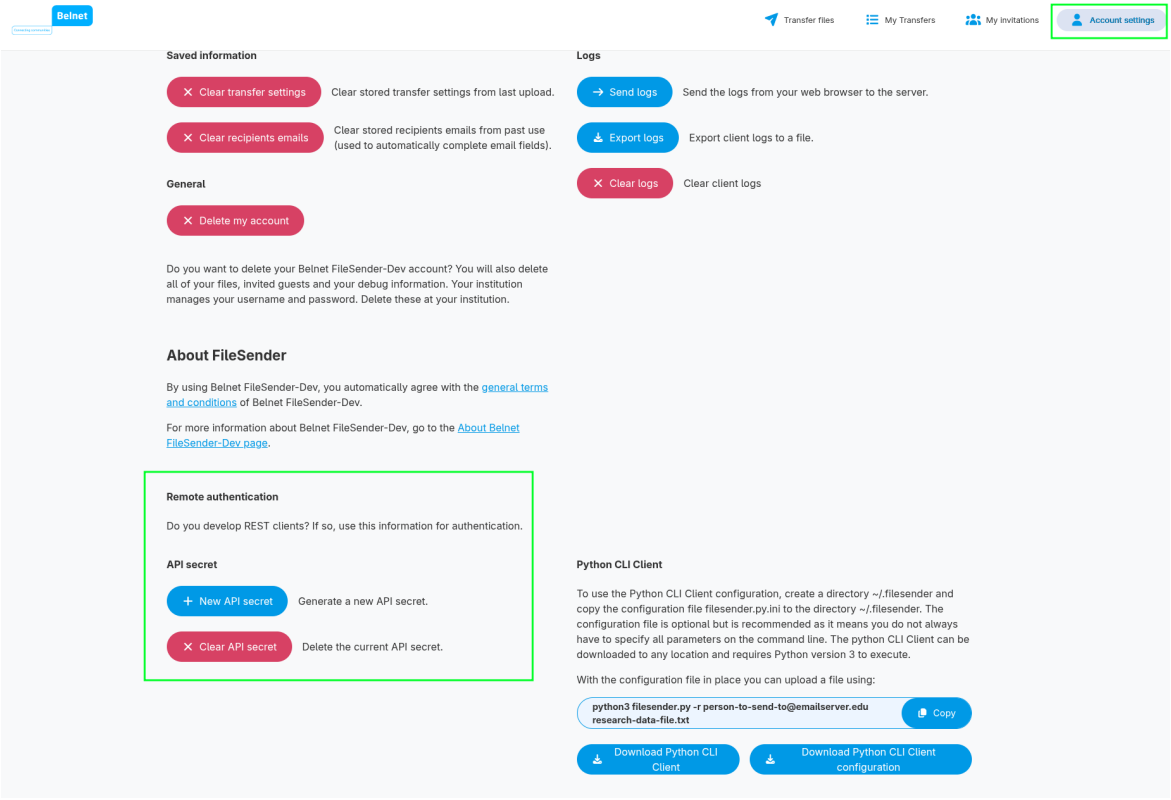
FileSender can generate a secret API key for you.
This key will be linked, within the system, to your user identity, as provided by your organisation's Identity Provider (IdP).

Belnet Filesender was migrated to branch 3 of the Filesender project on 05 Nov 2026 .

If you were already using the API prior to this migration, you will need to update your automations to use the new Python CLI client (filesender.py).

2. Secret & Identity

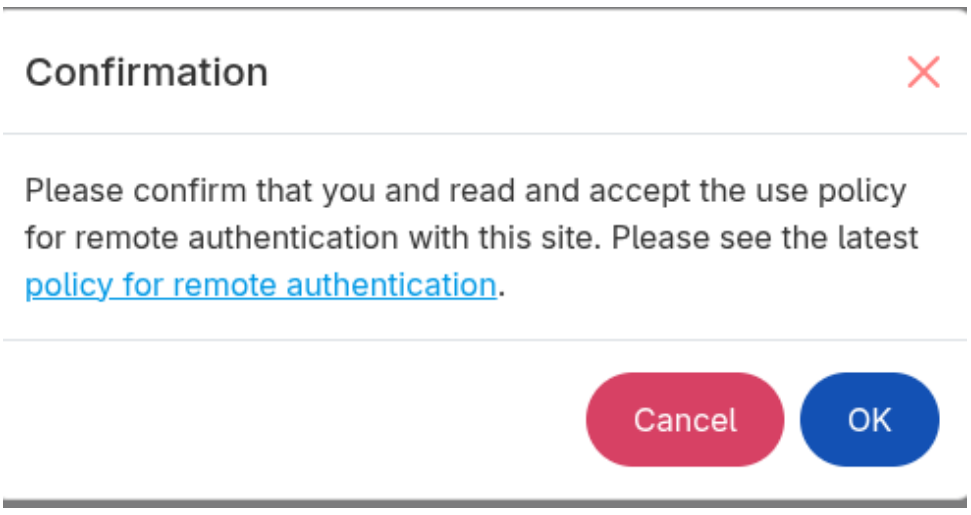
You can get/create/delete your API Secret key from the Filesender interface by clicking on the tab '**Account settings**' in the banner.
Users can delete current API secret and create a new one.



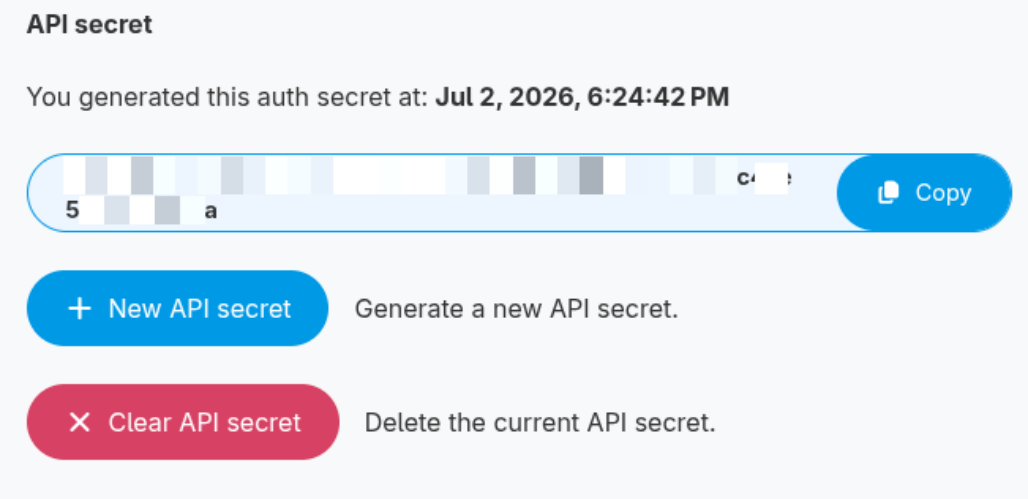
2.1.1. API secret

Click on "**New API secret**" to generate a new API secret or on "**Clear API secret**" if you already have one but want to replace it by creating a new one.

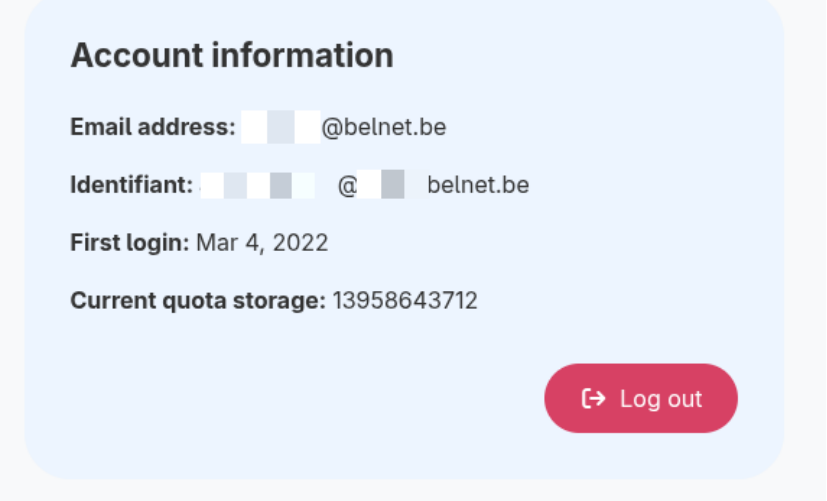
When creating a new API secret, the system will prompt you to confirm you've read and accept the use policy for metote authentication.



If the API key exists, it is displayed.
Its creation date is shown. A button allows you to copy it to your browser's clipboard.



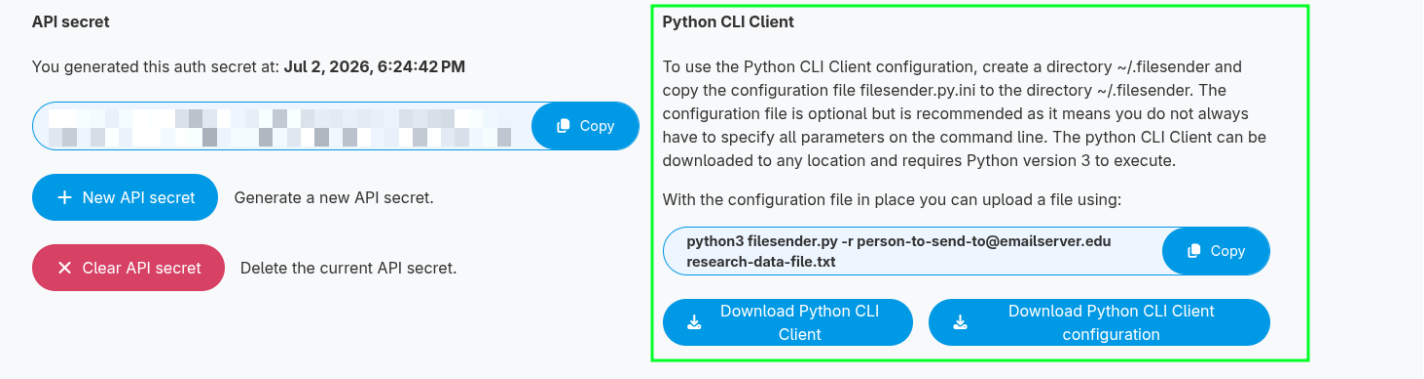
2.1.2. Identity



Under the 'Account Information' section, you will find your identity details provided by your organisation's identity provider (IdP) when you first logged in to the service.
It is these identity details that are stored in the database.
Your Identifiant (username) and your API key will be set automatically in the 'filesender.py.ini' file.

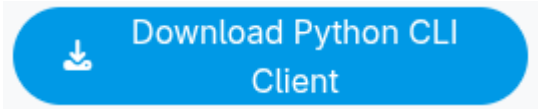
3. Python CLI Client

3.1. Python CLI Client files



- To use the Python CLI Client configuration (**filesender.py.ini)**, create a ~/.**filesender** directory (thus under your homedir).
- Python CLI Client (**filesender.py**), can be placed to any location and requires Python version 3 to execute.

CLI Client	File Name	Download button	Remarks	

Python CLI Client	filesender.py		• The Python CLI client for branch 3 of FileSender is now only available on the github project. When you click on the button, you will be redirected to the raw file "filesender.py"'. • Python CLI Client (**filesender.py**), can be placed to any location and requires Python version 3 to execute.	
Python CLI Client configuration	filesender.py.ini		• To use the Python CLI Client configuration (**filesender.py.ini)**, create a ~/.**filesender** directory (thus under your homedir). • The configuration file **filesender.py.ini** is optional but still recommended because it saves you from having to enter all the parameters on the command line. • When you download the Python CLI Client configuration file from the 'Account Settings' tab, it will **already contain all the minimal configuration details** relating to the system and your user account: ◦ base_url ◦ default_transfer_days_valid ◦ username (your identity) ◦ apikey (your API key if already created!)	

3.2. Usage example

Example of content for filesender.py.ini (Python CLI Client configuration)

filesender.py.ini (config file)

```
[system]
base_url = https://filesender.belnet.be/rest.php
default_transfer_days_valid = 7 # the number of days for which the transfer remains active, up to the maximum
number of days allowed by the service.

[user]
username = <your_identity>
apikey = <your_api_secret>
```

With the configuration file in place, you can create a Transfer (upload a file) with the command :

```
$ python3 filesender.py -v -m "Hello, here some file i'd share with you" -s "Sending using API" -u
<your_id>@<your_domain.tld> -a <mysecret> -r <recipient_addr@domain.tld> /Path/to/your/file_1.extension  /Path/to
/your/file_2.extension
```

Now you can simplify/shorten your command (not specify the -u & -a options):

```
$ python3 filesender.py -v -m "Hello, here some file i'd share with you" -s "Sending you file via API" -r
<recipient_addr@domain.tld> /Path/to/your/file_1.extension /Path/to/your/file_2.extension
```

default transfer days valid

By default, the transfer validity is 7 days. If more time is required, edit this line in the script filesender.py or overright this variable in the configuration file **filesender.py.ini**. For instance:

default_transfer_days_valid = 14

Constraints **Transfer expiry days (max)** & **Guest expiry days (max)** documented in the help section **remains valid also for the API use.**
Exceeding allowed max days will lead to PostTransfer failure !!



4. Conclusion

The script can be very useful to automate the uploading of files.

The transfers are visible on your profile if you login into the web interface:

- So you can extend, or delete the transfer.
- Although it should be possible to see and extend all transfer using Rest API.

Although there are a few disadvantages:

- The sender can be easily spoofed using the Rest API

5. Considerations

These scripts are those proposed by default by the Filesender opensource project. These scripts are not administered by Belnet.
Belnet will not propose any other script development for the use of Filesender APIs.

You are free to develop your own scripts according to your needs.

6. APi Documentation

You can find the documentation for FileSender's API at https://github.com/filesender/filesender/blob/master/docs/v3.0/rest/index.md.

Please note that this is an external project; not administered by Belnet.