

Batch File Encrypter

General

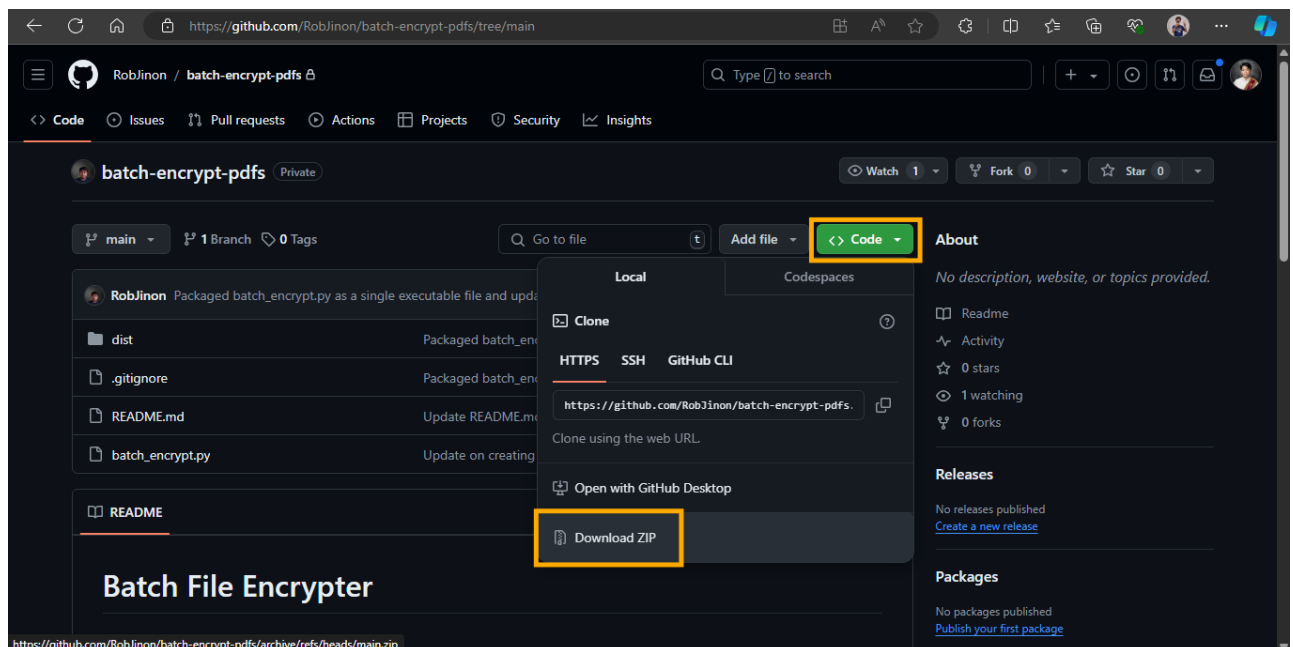
Developers: Robien Jinon, Limuelle Alamil

Description: This is a Python program that encrypts batch of files using **pikepdf** module. It asks for two inputs: 1) directory of files to be encrypted and 2) spreadsheet containing the data which will be used to encrypt the files.


[Download Batch File Encrypter v1](#)

? How to use (for end users)

1. Download the repository (<https://github.com/RobJinon/batch-encrypt-pdfs>). If you can't access the repository, ask for permission to be added as collaborator.









2. Extract the compressed repository and find the file **batch_encrypt.exe** inside the **dist** folder.

Name	Date modified	Type	Size
 batch_encrypt	7/9/2024 1:43 PM	Application	69,211 KB

3. Before starting, make sure that you have

- the files to be encrypted

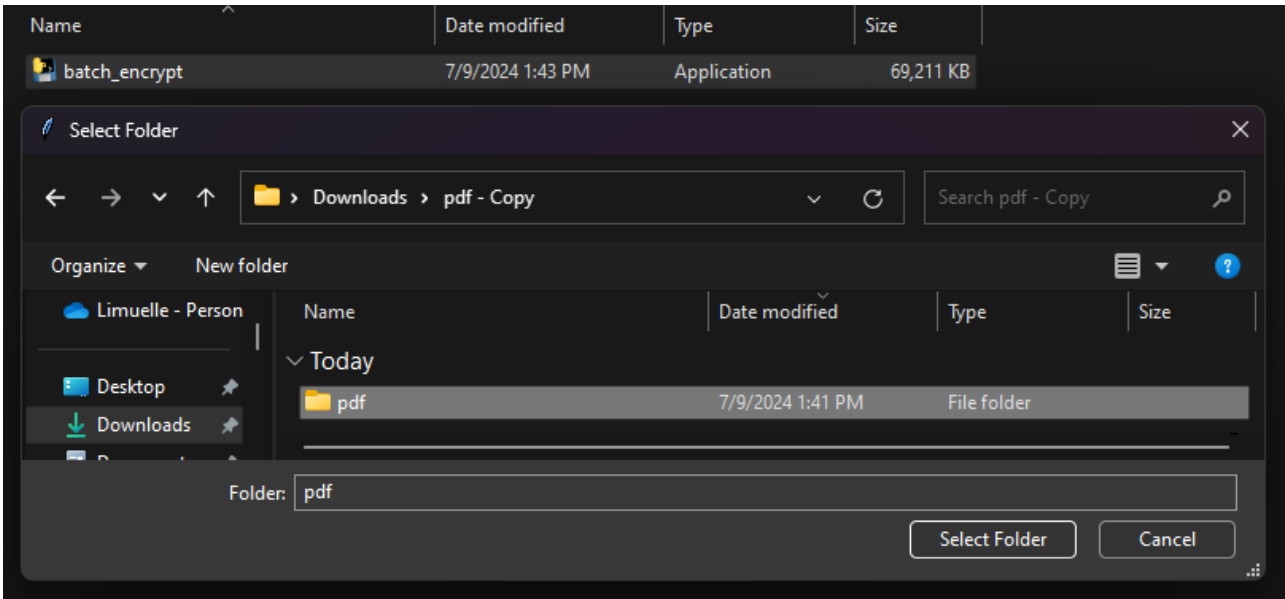
Name	Date modified	Type	Size
Yesterday			
 sample-list	7/8/2024 4:45 PM	Microsoft Excel W...	9 KB
 041015000123456	7/8/2024 4:43 PM	Microsoft Edge P...	372 KB
 041015000123589	7/8/2024 4:43 PM	Microsoft Edge P...	140 KB
 041015000123634	7/8/2024 4:43 PM	Microsoft Edge P...	164 KB
 041015000123764	7/8/2024 4:43 PM	Microsoft Edge P...	208 KB
 041015000123823	7/8/2024 4:43 PM	Microsoft Edge P...	162 KB

- the CSV/XLSX file that contains the list of filenames and birthdate ready; the filename refers to the file to be encrypted and the birthdate refers to its password

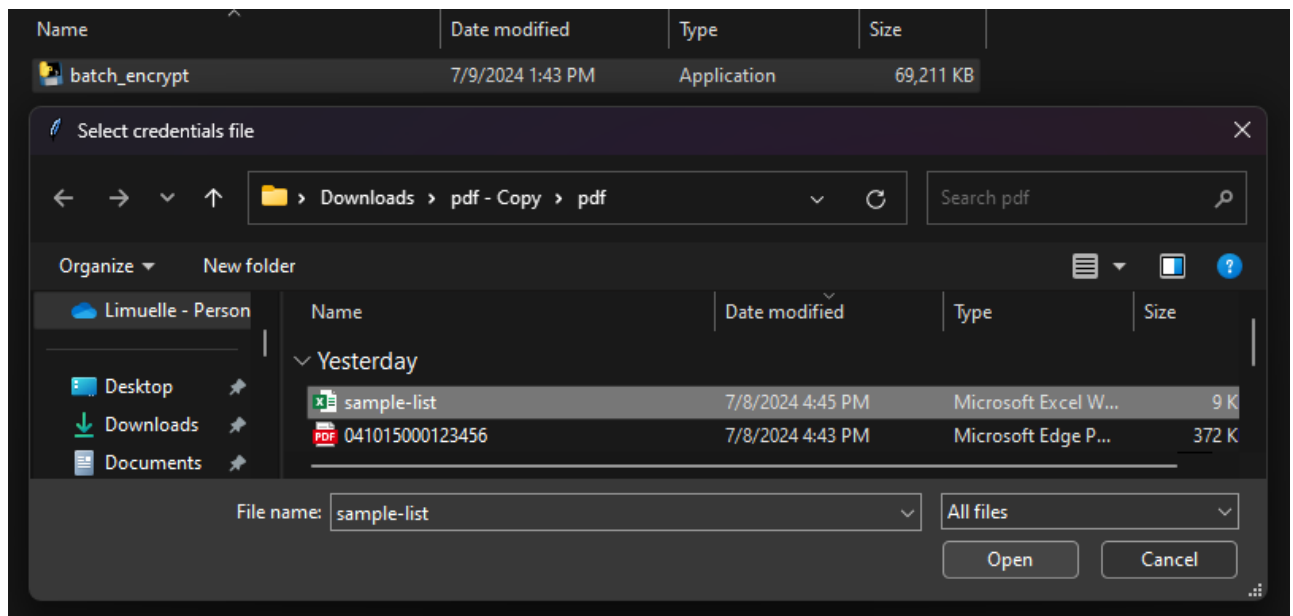
	A	B	C
1	Filename	Birthdate	
2	041015000123456	05041988	
3	041015000123589	05301987	
4	041015000123634	04051990	
5	041015000123764	10151978	
6	041015000123823	02281969	

- ⚠ NOTE: Make sure that the list of filenames in the CSV/XLSX file is coherent with the filenames of the actual files to be encrypted in the directory.
- ⚠ NOTE: Make sure that the filename and birthdate columns in the CSV/XLSX file have headers that are named "Filename" and "Birthdate" exactly (see image above) to avoid errors.
- ⚠ NOTE: Make sure that the files to be encrypted are all in the same directory. The CSV/XLSX file can be stored anywhere.

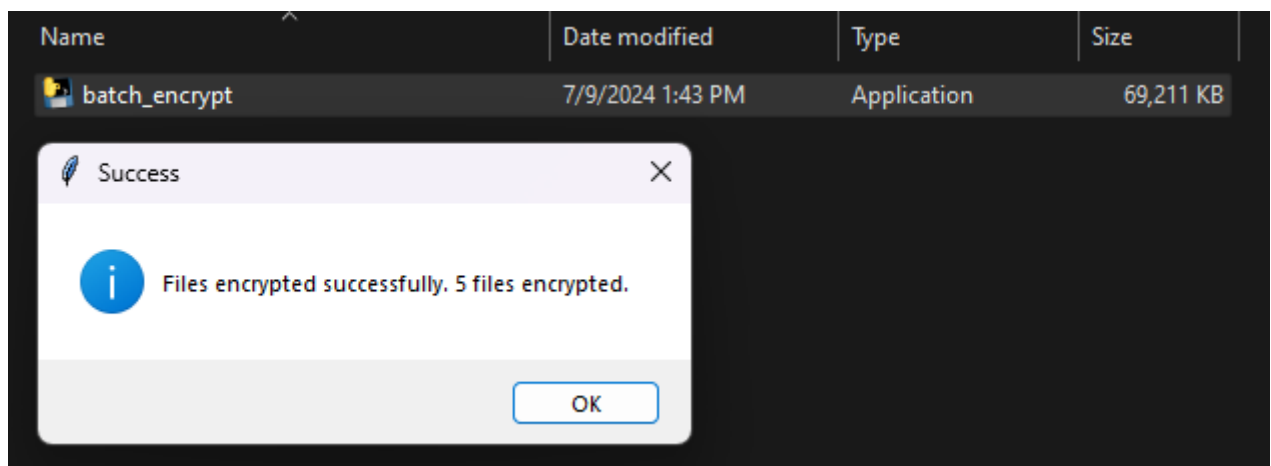
- 4. Double click the EXE file to start the program.
- 5. A file dialog asking to select a directory will appear. Select the directory where the files to be encrypted is located.



6. Another file dialog asking to select a CSV/XLSX file will appear. This CSV/XLSX file is where the list of filenames and birthdate are stored.



7. If the encryption is successful, a toast like the one below will appear. Click OK.



8. You can try to open the files to see if the encryption actually works.

? How to use (for developers)

1. Clone the repository (<https://github.com/RobJinon/batch-encrypt-pdfs>) to your local device. If you can't access, ask for permission to be added as collaborator.
2. The repository contains several files:
 - `.gitignore` - This file contains the files and directories that is not included when pushing changes to the repo.
 - `README.md` - This file contains documentation.
 - `batch_encrypt.py` - This Python file contains the source code for the program. Changes are made in this file.
 - `batch_encrypt.exe` - This file is the source code as an executable file.

3. If not yet installed, install the following modules first:

`pikepdf` - PDF encryption

`pandas` - CSV/XLSX file handling

4. To start making changes, open the repo and create a new branch by running `git branch`

`<branch_name>` in the bash terminal (e.g., `git branch lim-bug-fixes`). This way, you are not directly making modification in the working source code.

5. Once done with your changes, commit your changes and push your branch to the repository. If your branch has no merge conflicts with the `dev` branch (development branch), proceed to merge.

⚠ Note: Don't merge with the `main` branch unless you are absolutely sure that your branch works properly and error-free.