



Processor



Password-Protected PDFs



PROCESSOR &gt; GUIDES

# Manage password-protected PDFs seamlessly in Linux



PSPDFKit Processor has been deprecated and replaced by [Document Engine](#). To migrate to Document Engine and unlock advanced document processing capabilities, refer to our migration guide. Learn more about these enhancements on our [blog](#).

PSPDFKit Processor enables you to work with password-protected PDFs. You can provide password-protected PDFs as inputs for PSPDFKit Processor's operations and password protect the output of these operations.

Before you get started, make sure [Processor is up and running](#).

You can download and use either of the following sample documents for the examples in this guide:

- ✧ [Example eight-page PDF](#)
- ✧ [Example four-page PDF](#)

You'll be sending [multipart POST requests](#) with [instructions](#) to Processor's `/build` endpoint. To learn more about multipart requests, refer to our blog post on the topic, [A Brief Tour of Multipart Requests](#).

Check out the [API Reference](#) to learn more about the `/build` endpoint and all the actions you can perform on PDFs with PSPDFKit Processor.

## Performing Operations on Password-Protected



ASK AI

To merge a password-protected PDF with an unprotected PDF document, use the following example:

SHELL

HTTP

```
1 curl -X POST http://localhost:5000/api/build \
2   -F document=@/path/to/example-document.pdf \
3   -F password-document=@/path/to/example-password-document.pdf \
4   -F instructions='{
5     "parts": [
6       {
7         "file": "document"
8       },
9       {
10        "file": "password-document",
11        "password": "123456"
12      }
13    ]
14  }' \
15   -o result.pdf
```



The PDF documents you create with `/build` are unprotected by default. For more information on setting a password for a document, see the guide on [Output](#).

## Password Protecting the PDF Output

It's possible to protect PDF files with passwords to control who can access, modify, and print a document.

PDFs are protected with two types of passwords: the user password and the owner password.

- ✧ A user can open the document but they might have limited access to it depending on how the owner configures user permissions.
- ✧ An owner has full access to the document. They can change the PDF file, the user password, and the user permissions. These permissions control what the user can do with the file, such as view, print, copy, or modify the contents.

Specify what users can do to a password-protected document with the `user_permissions` parameter. The value of this parameter is an array, where each element is one of the following:

- ❖ `annotations_and_forms` allows users to add or modify text annotations and fill in interactive form fields.
- ❖ `assemble` allows users to insert, rotate, or delete pages and create document outline items or thumbnail images.
- ❖ `extract` allows users to copy or otherwise extract text and graphics from the document.
- ❖ `extract_accessibility` allows users to copy or otherwise extract text and graphics from the document using accessibility options.
- ❖ `fill_forms` allows users to fill in existing interactive form fields (including signature fields).
- ❖ `modification` allows users to modify the document in any way not covered by the other permissions settings.
- ❖ `print_high_quality` allows users to print the document in high quality.
- ❖ `printing` allows users to print the document.

To password protect the PDF output from `/build` , use the following example:

```
SHELL HTTP
1 curl -X POST http://localhost:5000/api/build \
2   -F cover=@/path/to/cover.pdf \
3   -F document=@/path/to/example-document.pdf \
4   -F instructions='{
5     "parts": [
6       {
7         "file": "cover"
8       },
9       {
10        "file": "document"
11      }
12    ],
13    "output": {
14      "type": "pdf",
15      "owner_password": "owner-password",
16      "user_password": "user-password",
17      "user_permissions": [
18        "printing",
19        "modification",
20        "extract",
21        "annotations_and_forms",
22        "fill_forms",
23        "extract_accessibility",
24        "assemble",
25        "print_high_quality"
26      ]
27    }
}
```

```
28  }' \
29  -o result.pdf
```

---

Was this helpful?

✓ YES

✗ NO

---

Questions? [Contact us](#)

