
open-pdf-sign

The `open-pdf-sign` CLI application allows to easily sign PDF files from the command line. Signatures can be invisible (default) or visible (can be customized).

Features

- Visible PDF signature in PDF (multi language support)
- Invoke via CLI or via starting a server
- Supported signature type: PAdES
- Supported signature profiles:
 - BASELINE-B
 - BASELINE-T
 - BASELINE-LT
 - BASELINE-LTA

Get Started

Download the latest JAR from the GitHub releases page or in your terminal:

```
1 curl --location --output open-pdf-sign.jar \  
2 https://github.com/open-pdf-sign/open-pdf-sign/releases/latest/  
   download/open-pdf-sign.jar
```

Alternatively, `open-pdf-sign` is also available on nix, a wrapper is available on npm, and alongside a installer for nginx.

Make sure that Java is installed in at least version 8.

Run

```
1 java -jar open-pdf-sign.jar \  
2 --input input.pdf --output output.pdf \  
3 --certificate certificate.crt --key keyfile.pem --passphrase  
   key_passphrase \  
4 --page -1 --locale de-AT
```

Usage:

```
1 Options:  
2 --add-page
```

```
3      add a blank page to the end of the document before signing
4  --baseline-lt
5      use PAdES profile with long-term validation material
6  --baseline-lta
7      use PAdES profile with long term availability and integrity of
      validation material
8  -b, --binary
9      binary output of PDF
10     Default: false
11  -c, --certificate
12     certificate (chain) to be used
13  --certification
14     Quality of signature certification (DocMDP) and allowed changes
      after
15     signing
16     Default: certified-minimal-changes-permitted
17     Possible Values: [not-certified, certified-no-change-permitted,
      certified-minimal-changes-permitted, certified-changes-
      permitted]
18  --config
19     use a configuration file
20  -h, --help
21     prints this page
22  --hint
23     text to be displayed in signature field
24  --host
25     run as server with the given hostname
26  --image
27     Image to be placed in signature block
28  --image-only
29     Only use the image as signature content
30     Default: false
31  -i, --input
32     input pdf file
33  -k, --key
34     signature key file or keystore
35  --label-hint
36     label for the 'hint' row
37  --label-signtee
38     label for the 'signtee' row
39  --label-timestamp
40     label for the 'timestamp' row
41  --left
42     X coordinate of the signature block in cm
43     Default: 1.0
44  -l, --locale
45     Locale, e.g. de-AT
46  --no-hint
47     don't display a hint row
48  -o, --output
49     output pdf file
```

```
50  --page
51      Page where the signature block should be placed. [-1] for last page
52  -p, --passphrase
53      passphrase for the signature key or keystore
54  --pdf-passphrase
55      Password required for reading a password-protected PDF input file
56  --port
57      run as server with the given port
58  --signature-contact
59      Contact information of the signer
60  --signature-location
61      The signer's location
62  --signature-reason
63      The signature creation reason
64  --timestamp
65      include signed timestamp
66      Default: false
67  --timezone
68      use specific timezone for time info, e.g. Europe/Vienna
69  --top
70      Y coordinate of the signature block in cm
71      Default: 1.0
72  --tsa
73      use specific time stamping authority as source (if multiple given,
74      will
75      be used in given order as fallback)
76      Default: []
77  --version
78      prints version of this program
79  --width
80      width of the signature block in cm
      Default: 10.0
```

Usage with Let's Encrypt certificates

PDFs can also be signed using your existing Let's Encrypt certificate.

```
1  java -jar open-pdf-sign.jar --input input.pdf --output output.pdf \
2    --certificate /etc/letsencrypt/live/openpdfsign.org/fullchain.pem \
3    --key /etc/letsencrypt/live/openpdfsign.org/privkey.pem
```

Signing documents with long-term validation info (PAdES-LT)

Sign documents with signatures that provides the long-term availability of the validation material by incorporating all the material or references to material required for validating the signature.

For this, using a timestamp is needed.

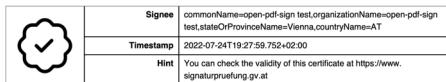
```
1 java -jar open-pdf-sign.jar --input input.pdf --output output.pdf \  
2   --certificate /etc/letsencrypt/live/openpdfsign.org/fullchain.pem \  
3   --key /etc/letsencrypt/live/openpdfsign.org/privkey.pem \  
4   --timestamp --tsa http://timestamp.digicert.com \  
5   --baseline-lt
```

Visible signatures

If the `page` parameter is specified, a visible signature will be placed on the specified page. For example, running

```
1 java -jar open-pdf-sign.jar --input input.pdf --output output.pdf \  
2   --certificate certificate.crt \  
3   --key key.pem \  
4   --page -1 --image mylogo.png \  
5   --hint "You can check the validity at https://www.signaturpruefung  
        .gv.at"
```

will place a visible signature looking similar to the image below on the last page (-1) of the PDF document.



Usage in server mode

You can also run open-pdf-sign as a server application in order to only load certificates once and easily integrate it in applications where CLI invocations are not possible. Simply add the `port` or `host` parameters, e.g.

```
1 java -jar open-pdf-sign.jar --input input.pdf --output output.pdf \  
2   --certificate /etc/letsencrypt/live/openpdfsign.org/fullchain.pem \  
3   --key /etc/letsencrypt/live/openpdfsign.org/privkey.pem \  
4   --port 8090 --host 127.0.0.1
```

Then, PDFs can be signed via the specified POST request:

```
1 curl --location 'http://localhost:8090/' \  
2   --header 'Content-Type: application/json' \  
3   --data-raw '{"input":"/path/to/pdf.pdf"}'
```

Using a config file

Instead of specifying everything via CLI parameters, you can also use a configuration file (e.g. this one):

```
1 java -jar open-pdf-sign.jar --config /path/to/config.yaml
```

This way, you could also configure multiple (virtual) hosts.

Development

Requirements

- Maven
- JDK 8

Build

```
1 mvn package
```

License

This project is licensed under the Apache 2.0-License.

The code contained in the org/openpdfsign/dss subfolder extends and modifies code from the dss project which is licensed under the LGPL-2.1 license.

This project received financial support from netidee.