

# The pdfprivacy package\*

Laurens Sion  
laurens@sion.info

May 24, 2026

## Abstract

Creating pdfs with pdfL<sup>A</sup>T<sub>E</sub>X populates several pdf meta-data fields such as date/time of creation/modification, information about the latex installation (e.g., pdfT<sub>E</sub>X version), and the relative paths of included pdfs. The pdfprivacy package provides support for emptying several of these pdf meta-data fields as well as suppressing some pdfT<sub>E</sub>X meta-data entries in the resulting pdf.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Requirements and Supported Engines</b>	<b>2</b>
2.1	pdfL <sup>A</sup> T <sub>E</sub> X (full support)	2
2.2	LuaL <sup>A</sup> T <sub>E</sub> X (full support)	2
2.3	XeL <sup>A</sup> T <sub>E</sub> X (limited support)	2
<b>3</b>	<b>Usage</b>	<b>3</b>
3.1	Options	3
3.2	Using pdfprivacy without hyperref	4
3.3	Removing the pdftrailerid	4
<b>4</b>	<b>Implementation</b>	<b>5</b>
4.1	Loading packages	5
4.2	Storage for options and settings	5
4.3	Declaring package options	5
4.4	Process the user preferences	6
4.5	Remove or suppress all the necessary meta-data	6
4.5.1	Processing pdftex	6
4.5.2	Processing xetex	7
4.5.3	Processing luatex	8

---

\*This document corresponds to pdfprivacy v1.1, dated 2026/05/24.

# 1 Introduction

By default, pdfs created by pdfL<sup>A</sup>T<sub>E</sub>X contain several meta-data fields populated with information about the application used to create them, date/time information, and information about the L<sup>A</sup>T<sub>E</sub>X installation. The pdfprivacy package provides support to remove several of these meta-data fields from the resulting pdf. By default pdfprivacy prevents the following properties from being set:

- The PDF creator (**Creator** in `hyperref`)
- The PDF producer (**Producer** in `hyperref`)
- The PDF modification date (**ModDate** in `hyperref`)
- The PDF creation date (**CreationDate** in `hyperref`)
- The PTEX.Fullbanner in the advanced PDF properties
- The PTEX.FileName with the filenames of any included pdf (e.g., images)
- The PTEX.PageNumber
- The PTEX.InfoDict

In addition to that, pdfprivacy can also remove the document meta-data: **Author**, **Title**, **Subject**, and **Keywords**; and the pdftrailerid which allows you to create reproducible pdfs.

## 2 Requirements and Supported Engines

pdfprivacy supports three T<sub>E</sub>X engines: pdfL<sup>A</sup>T<sub>E</sub>X, LuaL<sup>A</sup>T<sub>E</sub>X, and XeL<sup>A</sup>T<sub>E</sub>X. Support differs by engine due to availability of metadata suppression primitives. Note that there are caveats when you use certain primitives with certain engines (e.g., `\pdfinfo` appends entries to the dictionary instead of replacing them). See section 3.2 for more details.

### 2.1 pdfL<sup>A</sup>T<sub>E</sub>X (full support)

pdfprivacy requires pdfT<sub>E</sub>X version 1.40.17 or above<sup>1</sup> for full metadata removal and suppression of the pdfT<sub>E</sub>X banner:

```
PTEX.Fullbanner This is pdfTeX, Version 3.14159265...
```

All metadata removal options (`nodocdata`, `noproducerdata`, `noeditdata`, `noptexdata`, `nopdftrailerid`) are fully supported.

### 2.2 LuaL<sup>A</sup>T<sub>E</sub>X (full support)

LuaL<sup>A</sup>T<sub>E</sub>X provides the `\pdfvariable suppressoptionalinfo` primitive for comprehensive metadata suppression. All metadata removal options (`nodocdata`, `noproducerdata`, `noeditdata`, `noptexdata`, `nopdftrailerid`) are fully supported.

### 2.3 XeL<sup>A</sup>T<sub>E</sub>X (limited support)

XeL<sup>A</sup>T<sub>E</sub>X lacks several pdfT<sub>E</sub>X and LuaL<sup>A</sup>T<sub>E</sub>X primitives for metadata suppression. The following limitations apply:

---

<sup>1</sup><https://www.tug.org/texlive/Contents/live/texmf-dist/doc/pdftex/NEWS>

- `nodocdata` is *supported*: document metadata can be cleared via the `\special{pdf:docinfo}` mechanism.
- `noproducerdata` is *partially supported*: the PDF Creator can be cleared via `\special{pdf:docinfo}`, but clearing the Producer field is not possible as `xdvipdfmx` will override it.
- `noeditdata` is *not supported*: these require engine primitives unavailable in Xe<sub>La</sub>TeX. Using this options will raise a warning.
- `noptexdata` is *not supported*: Xe<sub>La</sub>TeX does not seem to emit PTEX meta-data entries in the resulting pdf, so there is nothing to suppress.
- `nopdftrailerid` is *partially supported*: instead of omitting the trailer ID, an all-zero ID `<00000000...00000000>` is inserted. This may affect workflows relying on unique trailer IDs.

### 3 Usage

To use `pdfprivacy`, simple include it:

```
\usepackage[<options>]{pdfprivacy},
```

Options are passed on to `pdfprivacy` as key-value pairs. By default, the options are set as follows:

```
nodocdata=false,
noeditdata=true,
noproducerdata=true,
noptexdata=true,
nopdftrailerid=false
```

This removes/suppresses all pdf creator, producer, dates, and pdfTeX meta-data. Standard document properties such as title, author, subject, and keywords are still kept. The `pdftrailerid` is kept as well by default.

#### 3.1 Options

All `pdfprivacy` options are boolean. Each option is prefixed with `no` to indicate you want the meta-data removed, you can specify the opposite as well (e.g., both `nodocdata=true` or `docdata=false` are valid options).

**`nodocdata=<boolean>`** Specify whether document meta-data should be removed from the pdf. Document meta-data includes: (i) Title, (ii) Subject, (iii) Author, and (iv) Keywords.

**`noproducerdata=<boolean>`** Specify whether meta-data on the pdf-producing application should be removed. This includes: (i) Creator, and (ii) Producer.

**`noeditdata=<boolean>`** Specify whether meta-data on the edit and creation dates should be removed. This includes: (i) creation date, and (ii) modification date.

**`noptexdata=<boolean>`** Specify whether the PTEX meta-data entries should be suppressed. This includes: (i) PTEX.Fullbanner, (ii) PTEX.FileName, (iii) PTEX.PageNumber, and (iv) PTEX.InfoDict.

**`nopdftrailerid=<boolean>`** Specify whether the pdftrailerid should be omitted.

`strict=<boolean>` Specify strict mode. In this mode, unsupported options will raise package errors instead of warnings, and the package will fail to load if the engine does not support the required primitives for the selected options.

### 3.2 Using pdfprivacy without hyperref

When using `pdfprivacy` without `hyperref`, the `\pdfinfo` command is used to set the pdf meta-data properties (i.e., `Title`, `Subject`, `Keywords`, `Author`, `Creator`, and `Producer`). The `\pdfinfo` works by appending the provided properties to the pdf info dictionary. This means that any previously set properties will still be present in the resulting pdf, despite pdf viewers only showing the last one. To make sure these properties are not present, *do not* manually set any of these properties, or use `hyperref` to set them. `Hyperref` will only save the last property (i.e., the empty one set by `pdfprivacy`) to the pdf. To make absolutely sure that no meta-data is present in the resulting pdf, you can always manually check the pdf by opening it as a text file.<sup>2</sup>

### 3.3 Removing the pdftrailerid

The pdf trailer ID is a generated ID to uniquely identify a pdf. The generated ID is based on the time of creation. This ID can be removed to make reproducible pdfs, i.e., with the same cryptographic hash.

**Important:** The pdf trailer id is kept by default because it is optional but strongly recommended by the pdf standard.<sup>3</sup> Not including this entry could break some workflows that rely on the trailer ID to uniquely identify files. If this is not a concern, the `nopdftrailerid=true` option can be used to remove it as well.

---

<sup>2</sup>Note: privacy-sensitive content may be present in the pdf in an encoded form. When manually verifying, make sure to look for the pdf property (e.g., `/Author (<content>)`) as well to verify it is empty.

<sup>3</sup>[https://www.adobe.com/content/dam/acom/en/devnet/acrobat/pdfs/pdf\\_reference\\_1-7.pdf#G8.1639448](https://www.adobe.com/content/dam/acom/en/devnet/acrobat/pdfs/pdf_reference_1-7.pdf#G8.1639448)

## 4 Implementation

### 4.1 Loading packages

Load `ifthen` for `ifthenelse` and `booleans` for managing user preferences.

```
1 \RequirePackage{ifthen}
```

Load `kvoptions` for loading key-value options.

```
2 \RequirePackage{kvoptions}
```

Load `iftex` to detect the TeX engine and provide clearer diagnostics.

```
3 \RequirePackage{iftex}
```

### 4.2 Storage for options and settings

Some `booleans` for keeping track if `hyperref` is loaded and if all meta-data should be removed.

```
4 \newboolean{pdfprivacy@hyperrefloaded}
5 \newboolean{pdfprivacy@removeall}
6 \setboolean{pdfprivacy@removeall}{false}
7 \newcommand{\pdfprivacy@enginename}{unknown engine}
8 \ifPDFTeX\renewcommand{\pdfprivacy@enginename}{pdfTeX}\fi
9 \ifXeTeX\renewcommand{\pdfprivacy@enginename}{XeTeX}\fi
10 \ifLuaTeX\renewcommand{\pdfprivacy@enginename}{LuaTeX}\fi
11 \newcommand{\pdfprivacy@unsupportedoptionerror}[2]{%
12 \ifthenelse{\boolean{pdfprivacy@strict}}{%
13 \PackageError{pdfprivacy}{Option '#1' is not supported with #2}{%
14 This option requires engine-specific PDF primitives that are unavailable.%
15 \MessageBreak Disable the option, use another engine, or use the default metadata behavior.%
16 }}{%
17 \PackageWarning{pdfprivacy}{Option '#1' is not supported with #2.
18 This option requires engine-specific PDF primitives that are unavailable.
19 Disable the option, use another engine, or use the default metadata behavior.
20 }}}
21 \newcommand{\pdfprivacy@warningorstrictfail}[2][{}]{%
22 \ifthenelse{\boolean{pdfprivacy@strict}}{%
23 \PackageError{pdfprivacy}{#2}{#1}}{%
24 \PackageWarning{pdfprivacy}{#2}}}
```

### 4.3 Declaring package options

Declare all the package options and their complementary ones.

```
25 \DeclareBoolOption[false]{nodocdata}
26 \DeclareComplementaryOption{docdata}{nodocdata}
27 \DeclareBoolOption[true]{noproducerdata}
28 \DeclareComplementaryOption{producerdata}{noproducerdata}
29 \DeclareBoolOption[true]{noeditdata}
30 \DeclareComplementaryOption{editdata}{noeditdata}
31 \DeclareBoolOption[true]{noptexdata}
32 \DeclareComplementaryOption{ptexdata}{noptexdata}
```

```

33 \DeclareBoolOption[false]{nopdftrailerid}
34 \DeclareComplementaryOption{pdftrailerid}{nopdftrailerid}
35 \DeclareBoolOption[false]{strict}

```

Additional `all` option to just remove everything.

```

36 \DeclareVoidOption{all}{\setboolean{pdfprivacy@removeall}{true}}

```

Process the options.

```

37 \ProcessKeyvalOptions*

```

## 4.4 Process the user preferences

Check if we need to remove `all` and override the other preferences.

```

38 \ifthenelse{\boolean{pdfprivacy@removeall}}{%
39 \setboolean{pdfprivacy@nodocdata}{true}
40 \setboolean{pdfprivacy@noproducerdata}{true}
41 \setboolean{pdfprivacy@noeditdata}{true}
42 \setboolean{pdfprivacy@noptexdata}{true}
43 \setboolean{pdfprivacy@nopdftrailerid}{true}
44 }{%
45 }

```

Check if `hyperref` is loaded as some meta-data needs to be set using `hyperref`.

```

46 \AtBeginDocument{
47 \@ifpackageloaded{hyperref}{\setboolean{pdfprivacy@hyperrefloaded}{true}}{\setboolean{pdfprivacy@hyperrefloaded}{false}}
48 }

```

## 4.5 Remove or suppress all the necessary meta-data

Process all preferences grouped by TeX engine. The implementation first dispatches on engine, then handles all option processing within the selected engine branch.

```

49 \AtBeginDocument{

```

### 4.5.1 Processing `pdftex`

In the `pdftex` branch, do removal and suppression with the `pdftex` primitives.

```

50 \ifpdftex

```

Process the document meta-data.

```

51 \ifthenelse{\boolean{pdfprivacy@nodocdata}}{%
52 \ifthenelse{\boolean{pdfprivacy@hyperrefloaded}}{%
53 \hypersetup{pdfinfo={ Author={}, Subject={}, Title={}, Keywords={} }}%
54 }{%
55 \pdfinfo{/Author () /Title () /Subject () /Keywords () }%
56 }
57 }{}

```

Process the producer meta-data.

```
58 \ifthenelse{\boolean{pdfprivacy@noproducerdata}}{%  
59   \ifthenelse{\boolean{pdfprivacy@hyperrefloaded}}{%  
60     \hypersetup{pdfinfo={ Creator={}, Producer={} }}%  
61   }{%  
62     \pdfinfo{/Creator () /Producer ()}%  
63   }  
64 }
```

Process the edit dates meta-data.

```
65 \ifthenelse{\boolean{pdfprivacy@noeditdata}}{%  
66   \pdfinfoomitdate1  
67 }
```

Process the ptex meta-data.

```
68 \ifthenelse{\boolean{pdfprivacy@noptexdata}}{%  
69   \pdfsuppressptexinfo-1  
70 }
```

Process the trailerid.

```
71 \ifthenelse{\boolean{pdfprivacy@nopdftrailerid}}{%  
72   \pdftrailerid{}  
73 }{}  
74
```

#### 4.5.2 Processing xetex

In the xetex branch, unsupported options produce package errors, while any primitive-backed behavior is retained if available.

```
75 \else\ifxetex
```

Process the document meta-data.

```
76 \ifthenelse{\boolean{pdfprivacy@nodocdata}}{%  
77   \ifthenelse{\boolean{pdfprivacy@hyperrefloaded}}{%  
78     \hypersetup{pdfinfo={ Author={}, Subject={}, Title={}, Keywords={} }}%  
79   }{%  
80     \special{pdf:docinfo << /Title () /Author () /Subject () /Keywords () >>}%  
81   }  
82 }
```

Process the producer meta-data.

```
83 \ifthenelse{\boolean{pdfprivacy@noproducerdata}}{%  
84   \pdfprivacy@warningorstrictfail[] {Clearing the PDF producer not supported with XeTeX. Onl  
85   \ifthenelse{\boolean{pdfprivacy@hyperrefloaded}}{%  
86     \hypersetup{pdfinfo={ Creator={}, Producer={} }}%  
87   }{%  
88     \special{pdf:docinfo << /Creator () /Producer () >>}%  
89   }  
90 }
```

```

91 \ifthenelse{\boolean{pdfprivacy@noeditdata}}{%
92   \pdfprivacy@unsupportedoptionerror{noeditdata}{XeTeX}
93 }{}
94 \ifthenelse{\boolean{pdfprivacy@noptexdata}}{%
95   \PackageInfo{pdfprivacy}{Option 'noptexdata' is not supported with XeTeX. XeTeX does not
96 }{}
97 \ifthenelse{\boolean{pdfprivacy@nopdftrailerid}}{%
98   \pdfprivacy@warningorstrictfail[]{}{No trailerid is not supported for XeTeX. Instead, an al
99 This might break some pdf workflows that rely on the trailerid.%
100 }
101   \special{pdf:trailerid [
102     <00000000000000000000000000000000>
103     <00000000000000000000000000000000>
104 ]}
105 }{}
106 \else\ifluatex

```

### 4.5.3 Processing luatex

In the Lua<sub>TEX</sub> branch, all metadata suppression is handled via `\pdfvariable suppressoptionalinfo` with a bitmask computed from the active options. Document metadata (nodoc-data) is cleared separately via `hyperref` or `\pdfinfo` since those fields are not covered by `\pdfvariable suppressoptionalinfo`.

Process the document meta-data.

```

107 \ifthenelse{\boolean{pdfprivacy@nodocdata}}{%
108   \ifthenelse{\boolean{pdfprivacy@hyperrefloaded}}{%
109     \hypersetup{pdfinfo={ Author={}, Subject={}, Title={}, Keywords={} }}%
110   }{}
111   \pdfextension info { /Title () /Author () /Subject () /Keywords () }
112 }
113 }{}

```

Compute the `\pdfvariable suppressoptionalinfo` bitmask from the active options and apply it. The bit values are:

- $1+2+4+8 = 15$ : PTEX.FullBanner, PTEX.FileName, PTEX.PageNumber, PTEX.InfoDict (noptexdata)
- $16+128 = 144$ : Creator, Producer (noproducerdata)
- $32+64 = 96$ : CreationDate, ModDate (noeditdata)
- $512$ : ID (nopdftrailerid)

When `hyperref` is loaded, Creator and Producer are also cleared via `\hypersetup` since `hyperref` writes those fields independently of the engine.

```

114 \newcount\pdfprivacy@suppressbits
115 \pdfprivacy@suppressbits=0
116 \ifthenelse{\boolean{pdfprivacy@noptexdata}}{\advance\pdfprivacy@suppressbits by 15}{}%
117 \ifthenelse{\boolean{pdfprivacy@noproducerdata}}{\advance\pdfprivacy@suppressbits by 144}{}%
118 \ifthenelse{\boolean{pdfprivacy@noeditdata}}{\advance\pdfprivacy@suppressbits by 96}{}%
119 \ifthenelse{\boolean{pdfprivacy@nopdftrailerid}}{\advance\pdfprivacy@suppressbits by 512}{}%
120 \pdfvariable suppressoptionalinfo\pdfprivacy@suppressbits
121 \ifthenelse{\boolean{pdfprivacy@noproducerdata}}{%
122   \ifthenelse{\boolean{pdfprivacy@hyperrefloaded}}{%

```



```

123     \hypersetup{pdfinfo={ Creator={}, Producer={} }}%
124     }{}%
125     {}
126 \else

```

As a fallback for unknown engines, enabled options are reported as unsupported using the generic engine name for diagnostics.

```

127 \ifthenelse{\boolean{pdfprivacy@nodocdata}}{%
128   \pdfprivacy@unsupportedoptionerror{nodocdata}{\pdfprivacy@enginename}
129 }{}
130 \ifthenelse{\boolean{pdfprivacy@noproducerdata}}{%
131   \pdfprivacy@unsupportedoptionerror{noproducerdata}{\pdfprivacy@enginename}
132 }{}
133 \ifthenelse{\boolean{pdfprivacy@noeditdata}}{%
134   \pdfprivacy@unsupportedoptionerror{noeditdata}{\pdfprivacy@enginename}
135 }{}
136 \ifthenelse{\boolean{pdfprivacy@noptexdata}}{%
137   \pdfprivacy@unsupportedoptionerror{noptexdata}{\pdfprivacy@enginename}
138 }{}
139 \ifthenelse{\boolean{pdfprivacy@nopdftrailerid}}{%
140   \pdfprivacy@unsupportedoptionerror{nopdftrailerid}{\pdfprivacy@enginename}
141 }{}
142 \fi\fi\fi
143 }

```