

Dangerzone

[Dangerzone](#) is a very useful tool to securely open [potentially dangerous files](#).

It supports more than 20 file types, including PDFs, all major office-suite formats, and the most common image types, which it can convert to safe PDFs.

“ [!tip] Tip {static} If you are in doubt whether you can safely open a file - just open it with Dangerzone!

Instructions on how to install Dangerzone can found on their website. They support all major operating systems.

How does it work?

Dangerzone destroys malware by rendering your document into pixels in a secure sandbox and reconstructing it locally as a PDF. Documents are sanitized in a sandbox with no network access, so if a malicious document can compromise the sandbox, it can't "phone home". The sandbox is based on container technology.

“ [!info] Info {static}

Dangerzone is a free and open source project, maintained by [Freedom of the Press Foundation \(FPF\)](#), a nonprofit organization that protects and defends press freedom.

In case you are not satisfied with the above explanation:

“ [!technical] Dangerzone "under-the-hood"

This information is from the [project's "about" page](#). Dangerzone uses Linux containers, which are isolated application environments that share the Linux kernel with their > host. On Windows and macOS, it uses Podman under the

hood, which spins containers in a dedicated virtual machine. Since > Dangerzone 0.10.0, all this complexity is hidden from the user. First, the sandbox:

1. Reads the original document from standard input
2. Uses LibreOffice or PyMuPDF to convert original document to a PDF
3. Uses PyMuPDF to split PDF into individual pages, and to convert those into RGB pixel data
4. Writes the number of pages and the RGB pixel data to its standard output Then that sandbox quits. The host reads the RGB pixel data from the container's standard output and:
5. If OCR is enabled, uses PyMuPDF to convert RGB pixel data into a compressed, searchable PDF
6. Otherwise uses PyMuPDF to convert RGB pixel data into a compressed, flat PDF
7. Stores the safe PDF in the specified directory with the -safe.pdf suffix, and archives the original one

Version #2

Erstellt: 2026-05-20 18:17:59 UTC von ESC-IT Migration Bot

Zuletzt aktualisiert: 2026-05-23 14:48:56 UTC von ESC-IT Migration Bot