Loop Bug 6-year life is again discovered, affecting most PDF viewers

According to German software developer Hanno Böck, an error was discovered in the PDF syntax library from 2011 to appear on the popular PDF viewers today.

According to German software developer Hanno Böck, an error was discovered in the PDF syntax library from 2011 to appear on the popular PDF viewers today.

The original bug affected the PDF syntax element on Evince, the document reader application for Linux. It was discovered by software developer Andreas Bogk, who also helped Evince fix the error and publish the results at the Chaos Communication Camp 2011 event.

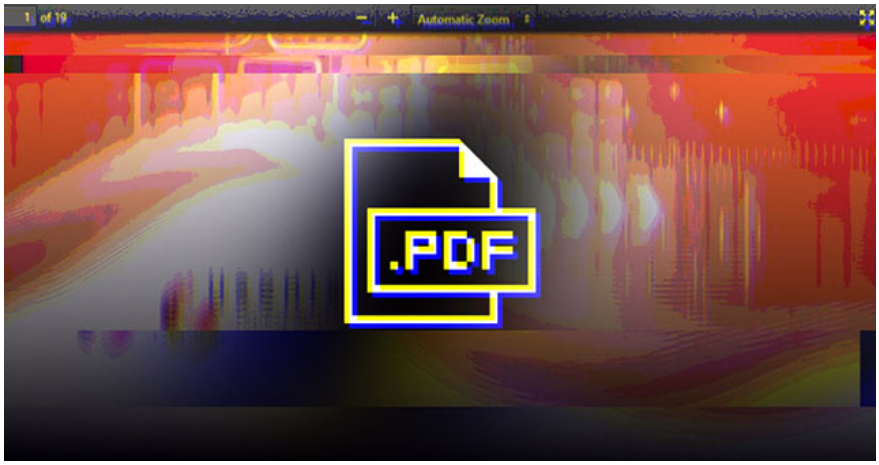
Bogk discovers certain structured PDF files - cross-references to the xref table - which will cause the Evince application to enter the continuous loop, taking up CPU resources and quickly filling up the memory and crashing.

This bug is mostly ignored because it is not considered a serious security bug and only affects a small application on a Linux machine.

Bug 6 years life back on the famous PDF Viewer

Six years later, the issue became more important after Böck discovered similar behavior in many popular PDF readers. Böck found this Loop Bug on PDFium, the library allows Chrome to render PDF files in the browser without a plugin.

The pdf.js library used in Firefox is also affected. Pdf.js is also used on GitHub to render PDF files on the website interface without users downloading files to view in 3rd party applications.



Error of loop error causes the application to crash

Windows Runtime PDF Renderer Library, WinRT PDF are all affected. This is the integrated PDF viewer of Edge and is also the default PDF syntax for Windows Reader App, the default PDF viewer on Windows 8 onwards.

Similarly, open source PDF syntax like Ghostscript and QPDF is also affected, meaning there will be many PDF viewing applications on the desktop and the web using the two tools affected. Böck reported a bug to affected products and the patch will be released soon.

Adobe Reader is not affected

Adobe Reader and the integrated PDF viewer on OS X are still safe.

The researcher said he used the fuzzing library for analysis. Fuzzing is a basic security testing technique that uses a lot of random input data and analyzes the output of the output and finds anomalies. Google's security experts use this technique very often and encourage people to use it.

Böck also blames administrators for not running test suites. This is a collection of problematic files that the viewer can still open without crashing. Most perfect, the software developer must not release a new version of the application without running a successful test suite.