New issue

Jump to bottom

Floating point exception on DCTStream::decodeImage #34

Open **strongcourage** opened this issue on May 29, 2019 · 0 comments

strongcourage commented on May 29, 2019

Hi,

Our fuzzer found a bug due to a floating point exception on the function DCTStream::decodeImage (the latest commit `b671b64` on master - version 0.70).

PoC: https://github.com/strongcourage/PoCs/blob/master/pdf2json_b671b64/PoC_fpe_DCTStream::decodeImage

Valgrind says:

```
valgrind pdf2json $PoC /dev/null
==3166== Memcheck, a memory error detector
==3166== Copyright (C) 2002-2015, and GNU GPL'd, by Julian Seward et al.
==3166== Using Valgrind-3.11.0 and LibVEX; rerun with -h for copyright info
==3166== Command: ./pdf2json ./PoC_fpe_DCTStream::decodeImage /dev/null
==3166==
Error (13268): Command token too long
Error (13372): Illegal character '>'
Error: PDF file is damaged - attempting to reconstruct xref table...
Error: End of file inside array
Error: End of file inside dictionary
Error (154): Dictionary key must be a name object
Error (165): Dictionary key must be a name object
Error (528): Dictionary key must be a name object
Error (530): Dictionary key must be a name object
Error (532): Dictionary key must be a name object
Error (536): Dictionary key must be a name object
Error (539): Dictionary key must be a name object
Error (545): Dictionary key must be a name object
Error (7892): Missing 'endstream'
Error (12313): Bad DCT data: missing 00 after ff
Error (12887): Bad DCT header
==3166==
==3166== Process terminating with default action of signal 8 (SIGFPE)
==3166==  Integer divide by zero at address 0x802EBDD05
==3166==    at 0x43533F: DCTStream::decodeImage() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==3166==    by 0x432C6C: DCTStream::reset() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==3166==    by 0x40941E: Object::streamReset() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==3166==    by 0x48788A: Lexer::Lexer(XRef*, Object*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==3166==    by 0x4542CE: Gfx::display(Object*, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==3166==    by 0x42A59E: Page::displaySlice(OutputDev*, double, double, int, int, int, int, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in
/home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==3166==    by 0x42A14A: Page::display(OutputDev*, double, double, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==3166==    by 0x42BCBD: PDFDoc::displayPage(OutputDev*, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==3166==    by 0x42BD48: PDFDoc::displayPages(OutputDev*, int, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==3166==    by 0x40269A: main (pdf2json.cc:275)
==3166==
==3166== HEAP SUMMARY:
==3166==     in use at exit: 263,639 bytes in 1,768 blocks
==3166==   total heap usage: 1,973 allocs, 205 frees, 355,024 bytes allocated
==3166==
==3166== LEAK SUMMARY:
==3166==    definitely lost: 16 bytes in 1 blocks
==3166==    indirectly lost: 8 bytes in 1 blocks
==3166==      possibly lost: 0 bytes in 0 blocks
==3166==    still reachable: 263,615 bytes in 1,766 blocks
==3166==         suppressed: 0 bytes in 0 blocks
==3166== Rerun with --leak-check=full to see details of leaked memory
==3166==
==3166== For counts of detected and suppressed errors, rerun with: -v
==3166== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
Floating point exception
```

Thanks,
Manh Dung

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant