New issue

Jump to bottom

SEGV on DCTStream::reset #30

Open **strongcourage** opened this issue on May 28, 2019 · 0 comments

strongcourage commented on May 28, 2019

Hi,

Our fuzzer found a crash due to an invalid write on the function DCTStream::reset (the latest commit `b671b64` on master - version 0.70).

PoC: https://github.com/strongcourage/PoCs/blob/master/pdf2json_b671b64/PoC_segv_DCTStream::reset

Valgrind says:

```
valgrind pdf2json $PoC /dev/null
==5144== Memcheck, a memory error detector
==5144== Copyright (C) 2002-2015, and GNU GPL'd, by Julian Seward et al.
==5144== Using Valgrind-3.11.0 and LibVEX; rerun with -h for copyright info
==5144== Command: ./pdf2json ./PoC_segv_DCTStream::reset /dev/null
==5144==
Error (13145): Illegal character ')'
Error: PDF file is damaged - attempting to reconstruct xref table...
Error (7397): Dictionary key must be a name object
Error (7407): Dictionary key must be a name object
Error (7408): Illegal character '>'
Error (7408): Dictionary key must be a name object
Error (7416): Dictionary key must be a name object
Error (7422): Dictionary key must be a name object
Error (7905): Dictionary key must be a name object
Error (7955): Dictionary key must be a name object
Error (7965): Dictionary key must be a name object
Error (7972): Dictionary key must be a name object
Error (7974): Dictionary key must be a name object
Error (7976): Dictionary key must be a name object
Error (7980): Dictionary key must be a name object
Error (7985): Dictionary key must be a name object
Error (7992): Dictionary key must be a name object
Error (7995): Dictionary key must be a name object
Error (7997): Dictionary key must be a name object
Error (8001): Dictionary key must be a name object
Error (8012): Dictionary key must be a name object
Error (8024): Dictionary key must be a name object
Error (8028): Dictionary key must be a name object
Error (8031): Dictionary key must be a name object
Error (8039): Dictionary key must be a name object
==5144== Invalid write of size 8
==5144==    at 0x432D2D: DCTStream::reset() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x40941E: Object::streamReset() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x48788A: Lexer::Lexer(XRef*, Object*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x4542CE: Gfx::display(Object*, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A59E: Page::displaySlice(OutputDev*, double, double, int, int, int, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in
/home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A14A: Page::display(OutputDev*, double, double, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BCBD: PDFDoc::displayPage(OutputDev*, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BD48: PDFDoc::displayPages(OutputDev*, int, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x40269A: main (pdf2json.cc:275)
==5144==  Address 0x5b1b218 is 0 bytes after a block of size 4,584 alloc'd
==5144==    at 0x4C2E0EF: operator new(unsigned long) (in /usr/lib/valgrind/vgpreload_memcheck-amd64-linux.so)
==5144==    by 0x42CF21: Stream::makeFilter(char*, Stream*, Object*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42C777: Stream::addFilters(Object*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x489AFF: Parser::makeStream(Object*, unsigned char*, CryptAlgorithm, int, int, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x489549: Parser::getObj(Object*, unsigned char*, CryptAlgorithm, int, int, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x43FA44: XRef::fetch(int, int, Object*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x4284A9: Object::fetch(XRef*, Object*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A565: Page::displaySlice(OutputDev*, double, double, int, int, int, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in
/home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A14A: Page::display(OutputDev*, double, double, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BCBD: PDFDoc::displayPage(OutputDev*, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BD48: PDFDoc::displayPages(OutputDev*, int, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x40269A: main (pdf2json.cc:275)
==5144==
==5144== Conditional jump or move depends on uninitialised value(s)
==5144==    at 0x436A29: DCTStream::readHuffSym(DCTHuffTable*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x434B49: DCTStream::readDataUnit(DCTHuffTable*, DCTHuffTable*, int*, int*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x4335CF: DCTStream::readMCURow() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x432FB6: DCTStream::getChar() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x40947A: Object::streamGetChar() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x48796F: Lexer::getChar() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x487A83: Lexer::getObj(Object*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x4890DF: Parser::Parser(XRef*, Lexer*, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x4542F8: Gfx::display(Object*, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A59E: Page::displaySlice(OutputDev*, double, double, int, int, int, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in
/home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A14A: Page::display(OutputDev*, double, double, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BCBD: PDFDoc::displayPage(OutputDev*, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==
Error (8504): Bad Huffman code in DCT stream
==5144== Use of uninitialised value of size 8
==5144==    at 0x41ACFE: GfxState::~GfxState() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x41C839: GfxState::restore() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x467BEB: Gfx::restoreState() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A5AA: Page::displaySlice(OutputDev*, double, double, int, int, int, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in
/home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A14A: Page::display(OutputDev*, double, double, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BCBD: PDFDoc::displayPage(OutputDev*, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BD48: PDFDoc::displayPages(OutputDev*, int, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x40269A: main (pdf2json.cc:275)
==5144==
==5144== Invalid read of size 8
```

```
==5144==    at 0x41ACFE: GfxState::~GfxState() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x41C839: GfxState::restore() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x467BEB: Gfx::restoreState() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A5AA: 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)
==5144==    by 0x42A14A: Page::display(OutputDev*, double, double, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BCBD: PDFDoc::displayPage(OutputDev*, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BD48: PDFDoc::displayPages(OutputDev*, int, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x40269A: main (pdf2json.cc:275)
==5144==  Address 0x8 is not stack'd, malloc'd or (recently) free'd
==5144==
==5144==
==5144== Process terminating with default action of signal 11 (SIGSEGV)
==5144==  Access not within mapped region at address 0x8
==5144==    at 0x41ACFE: GfxState::~GfxState() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x41C839: GfxState::restore() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x467BEB: Gfx::restoreState() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42A5AA: 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)
==5144==    by 0x42A14A: Page::display(OutputDev*, double, double, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BCBD: PDFDoc::displayPage(OutputDev*, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x42BD48: PDFDoc::displayPages(OutputDev*, int, int, double, double, int, int, int, int, int (*)(void*), void*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==5144==    by 0x40269A: main (pdf2json.cc:275)
==5144==  If you believe this happened as a result of a stack
==5144==  overflow in your program's main thread (unlikely but
==5144==  possible), you can try to increase the size of the
==5144==  main thread stack using the --main-stacksize= flag.
==5144==  The main thread stack size used in this run was 8388608.
==5144==
==5144== HEAP SUMMARY:
==5144==     in use at exit: 246,401 bytes in 1,945 blocks
==5144==   total heap usage: 2,354 allocs, 409 frees, 378,948 bytes allocated
==5144==
==5144== LEAK SUMMARY:
==5144==    definitely lost: 18,992 bytes in 161 blocks
==5144==    indirectly lost: 8 bytes in 1 blocks
==5144==      possibly lost: 0 bytes in 0 blocks
==5144==    still reachable: 227,401 bytes in 1,783 blocks
==5144==         suppressed: 0 bytes in 0 blocks
==5144== Rerun with --leak-check=full to see details of leaked memory
==5144==
==5144== For counts of detected and suppressed errors, rerun with: -v
==5144== Use --track-origins=yes to see where uninitialised values come from
==5144== ERROR SUMMARY: 27 errors from 4 contexts (suppressed: 0 from 0)
Segmentation fault
```

Thanks,
Manh Dung

strongcourage changed the title ~~**Segmentation fault on DCTStream::reset**~~ **SEGV on DCTStream::reset** on May 29, 2019

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant