New issue

Jump to bottom

SEGV on DCTStream::transformDataUnit #27

Open **strongcourage** opened this issue on May 27, 2019 · 0 comments

strongcourage commented on May 27, 2019 • edited

Hi,

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

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

Valgrind says:

```
valgrind pdf2json $PoC /dev/null
...
==23848== Invalid read of size 1
==23848==    at 0x43698B: DCTStream::transformDataUnit(unsigned short*, int*, unsigned char*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x43363B: DCTStream::readMCURow() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x432FB6: DCTStream::getChar() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x40947A: Object::streamGetChar() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x48796F: Lexer::getChar() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x487A83: Lexer::getObj(Object*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x489C19: Parser::shift() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x489825: Parser::getObj(Object*, unsigned char*, CryptAlgorithm, int, int, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x454759: Gfx::go(int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x454311: Gfx::display(Object*, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    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)
==23848==    by 0x42A14A: Page::display(OutputDev*, double, double, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in
/home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==  Address 0x6fd636 is not stack'd, malloc'd or (recently) free'd
==23848==
==23848==
==23848== Process terminating with default action of signal 11 (SIGSEGV)
==23848==  Access not within mapped region at address 0x6FD636
==23848==    at 0x43698B: DCTStream::transformDataUnit(unsigned short*, int*, unsigned char*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x43363B: DCTStream::readMCURow() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x432FB6: DCTStream::getChar() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x40947A: Object::streamGetChar() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x48796F: Lexer::getChar() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x487A83: Lexer::getObj(Object*) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x489C19: Parser::shift() (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x489825: Parser::getObj(Object*, unsigned char*, CryptAlgorithm, int, int, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x454759: Gfx::go(int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    by 0x454311: Gfx::display(Object*, int) (in /home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==    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)
==23848==    by 0x42A14A: Page::display(OutputDev*, double, double, int, int, int, Links*, int, Catalog*, int (*)(void*), void*) (in
/home/dungnguyen/PoCs/pdf2json_b671b64/pdf2json)
==23848==  If you believe this happened as a result of a stack
==23848==  overflow in your program's main thread (unlikely but
==23848==  possible), you can try to increase the size of the
==23848==  main thread stack using the --main-stacksize= flag.
==23848==  The main thread stack size used in this run was 8388608.
==23848==
==23848== HEAP SUMMARY:
==23848==     in use at exit: 230,627 bytes in 1,819 blocks
==23848==   total heap usage: 2,242 allocs, 423 frees, 357,101 bytes allocated
==23848==
==23848== LEAK SUMMARY:
==23848==    definitely lost: 16 bytes in 1 blocks
==23848==    indirectly lost: 8 bytes in 1 blocks
==23848==      possibly lost: 0 bytes in 0 blocks
==23848==    still reachable: 230,603 bytes in 1,817 blocks
==23848==         suppressed: 0 bytes in 0 blocks
==23848== Rerun with --leak-check=full to see details of leaked memory
==23848==
==23848== For counts of detected and suppressed errors, rerun with: -v
==23848== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
Segmentation fault
```

Thanks,
Manh Dung

strongcourage changed the title ~~Segmentation fault on DCTStream::transformDataUnit~~ SEGV on DCTStream::transformDataUnit 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