Another heap buffer overflow in libopenjp2 #1231

Closed sebastianpoeplau opened this issue on Jan 28, 2020 · 2 comments · Fixed by #1232

sebastianpoeplau commented on Jan 28, 2020 • edited

Hi,

This overflow looks similar to #1228 but still works on latest master (`b63a433`):

```
$ build_afl/bin/opj_decompress -i ../openjpeg/afl_symcc_5_out/afl-slave/crashes/id:000000,sig:06,sync:symcc,src:002975 -o /tmp/image_verification.pgm


===========================================
The extension of this file is incorrect.
FOUND 2975. SHOULD BE .jp2
===========================================

[INFO] Start to read j2k main header (884).
[INFO] Main header has been correctly decoded.
[INFO] No decoded area parameters, set the decoded area to the whole image
[INFO] Header of tile 1 / 1 has been read.
=================================================================
==3010==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x7fb19e1f8918 at pc 0x7fb1a258d7e9 bp 0x7ffd68100b90 sp 0x7ffd68100b88
WRITE of size 16 at 0x7fb19e1f8918 thread T0
    #0 0x7fb1a258d7e8 in opj_t1_clbl_decode_processor /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/t1.c:1765:73
    #1 0x7fb1a2441e05 in opj_thread_pool_submit_job /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/thread.c:835:9
    #2 0x7fb1a256f753 in opj_t1_decode_cblks /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/t1.c:1901:21
    #3 0x7fb1a263ca3c in opj_tcd_t1_decode /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/tcd.c:1969:9
    #4 0x7fb1a263ca3c in opj_tcd_decode_tile /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/tcd.c:1623
    #5 0x7fb1a24bfe91 in opj_j2k_decode_tile /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/j2k.c:8932:11
    #6 0x7fb1a24fd969 in opj_j2k_decode_tiles /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/j2k.c:10763:15
    #7 0x7fb1a24cd165 in opj_j2k_exec /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/j2k.c:8090:33
    #8 0x7fb1a24cd165 in opj_j2k_decode /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/j2k.c:11066
    #9 0x7fb1a2517dae in opj_jp2_decode /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/jp2.c:1603:11
    #10 0x512f27 in main /home/seba/tested_software/openjpeg_patched/src/bin/jp2/opj_decompress.c:1542:19
    #11 0x7fb1a20b709a in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x2409a)
    #12 0x42f719 in _start (/home/seba/tested_software/openjpeg_patched/build_afl/bin/opj_decompress+0x42f719)

0x7fb19e1f8918 is located 0 bytes to the right of 9998616-byte region [0x7fb19d86f800,0x7fb19e1f8918)
allocated by thread T0 here:
    #0 0x4dbd99 in __interceptor_posix_memalign (/home/seba/tested_software/openjpeg_patched/build_afl/bin/opj_decompress+0x4dbd99)
    #1 0x7fb1a265b9c7 in opj_aligned_alloc_n /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/opj_malloc.c:61:9
    #2 0x7fb1a265b9c7 in opj_aligned_malloc /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/opj_malloc.c:209
    #3 0x7fb1a263af32 in opj_alloc_tile_component_data /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/tcd.c:694:39
    #4 0x7fb1a263af32 in opj_tcd_decode_tile /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/tcd.c:1530
    #5 0x7fb1a24bfe91 in opj_j2k_decode_tile /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/j2k.c:8932:11
    #6 0x7fb1a24fd969 in opj_j2k_decode_tiles /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/j2k.c:10763:15
    #7 0x7fb1a24cd165 in opj_j2k_exec /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/j2k.c:8090:33
    #8 0x7fb1a24cd165 in opj_j2k_decode /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/j2k.c:11066

SUMMARY: AddressSanitizer: heap-buffer-overflow /home/seba/tested_software/openjpeg_patched/src/lib/openjp2/t1.c:1765:73 in opj_t1_clbl_decode_processor
Shadow bytes around the buggy address:
  0x0ff6b3c370d0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0ff6b3c370e0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0ff6b3c370f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0ff6b3c37100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0ff6b3c37110: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0ff6b3c37120: 00 00 00[fa]fa fa fa fa fa fa fa fa fa fa fa fa
  0x0ff6b3c37130: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0ff6b3c37140: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0ff6b3c37150: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0ff6b3c37160: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0ff6b3c37170: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
Shadow byte legend (one shadow byte represents 8 application bytes):
  Addressable:           00
  Partially addressable: 01 02 03 04 05 06 07
  Heap left redzone:       fa
  Freed heap region:       fd
  Stack left redzone:      f1
  Stack mid redzone:       f2
  Stack right redzone:     f3
  Stack after return:      f5
  Stack use after scope:   f8
  Global redzone:          f9
  Global init order:       f6
  Poisoned by user:        f7
  Container overflow:      fc
  Array cookie:            ac
  Intra object redzone:    bb
  ASan internal:           fe
  Left alloca redzone:     ca
  Right alloca redzone:    cb
  Shadow gap:              cc
==3010==ABORTING
```

Steps to reproduce as in #1228; the crashing input is available here.

Thank you!

sebastianpoeplau commented on Jan 29, 2020 Author

CVE-2020-8112 has been assigned.

rouault added a commit to rouault/openjpeg that referenced this issue on Jan 29, 2020

`opj_tcd_init_tile(): avoid integer overflow` ··· `05f9b91`

rouault mentioned this issue on Jan 29, 2020

opj_tcd_init_tile(): avoid integer overflow #1232

Merged

rouault closed this as completed in #1232 on Jan 30, 2020

---

sebastianpoeplau commented on Jan 30, 2020 Author

Thank you!

This was referenced on Mar 12, 2020

openjpeg: patch CVE-2020-6851 and CVE-2020-8112 NixOS/nixpkgs#82426

Merged

[20.03] openjpeg: patch CVE-2020-6851 and CVE-2020-8112 NixOS/nixpkgs#82444

Merged

[19.09] openjpeg: patch CVE-2020-6851 and CVE-2020-8112 NixOS/nixpkgs#82445

Merged

mtremer pushed a commit to ipfire/ipfire-2.x that referenced this issue on Apr 29

`openjpeg: Update to version 2.4.0` ··· `ca98d29`

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

opj_tcd_init_tile(): avoid integer overflow
rouault/openjpeg

1 participant