# jp2_decode() heap-buffer-overflow vulnerability #264

Closed dgh05t opened this issue on Jan 29, 2021 · 5 comments

**dgh05t** commented on Jan 29, 2021

Hi,
there's a heap-buffer-overflow vulnerability in function jp2_decode() , ( jp2_dec.c:280 )
poc: poc.zip
please compile the Jasper with ASAN, and run the poc with `"./jasper -f ~/Desktop/poc.jp2 --output-format jpg"` .

It seems because of the `ata.bpcc.bpcs` is not equal with `image->bpcs`

```
==62885==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x602000000254 at pc 0x7f2781d31c39 bp 0x7ffffae212a0 sp 0x7ffffae21290
READ of size 1 at 0x602000000254 thread T0
    #0 0x7f2781d31c38 in jp2_decode /home/dgh05t/fuzz/jasper-master/src/libjasper/jp2/jp2_dec.c:280
    #1 0x7f2781cf90e4 in jas_image_decode /home/dgh05t/fuzz/jasper-master/src/libjasper/base/jas_image.c:436
    #2 0x557083f77b62 in main /home/dgh05t/fuzz/jasper-master/src/appl/jasper.c:217
    #3 0x7f2781ad40b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)
    #4 0x557083f7874d in _start (/home/dgh05t/fuzz/jasper-master/build/src/appl/jasper+0x574d)

0x602000000254 is located 0 bytes to the right of 4-byte region [0x602000000250,0x602000000254)
allocated by thread T0 here:
    #0 0x7f2781f79bc8 in malloc (/lib/x86_64-linux-gnu/libasan.so.5+0x10dbc8)
    #1 0x7f2781d04886 in jas_malloc /home/dgh05t/fuzz/jasper-master/src/libjasper/base/jas_malloc.c:238

SUMMARY: AddressSanitizer: heap-buffer-overflow /home/dgh05t/fuzz/jasper-master/src/libjasper/jp2/jp2_dec.c:280 in jp2_decode
Shadow bytes around the buggy address:
  0x0c047fff7ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff8000: fa fa 04 fa fa fa fa 04 fa fa fa 04 fa fa fa 04
  0x0c047fff8010: fa fa 04 fa fa fa 04 fa fa fa 04 fa fa fa 04 fa
  0x0c047fff8020: fa fa 04 fa fa fa 04 fa fa fa 04 fa fa fa 04 fa
  0x0c047fff8030: fa fa 04 fa fa fa 04 fa fa fa 04 fa fa fa 04 fa
=>0x0c047fff8040: fa fa 04 fa fa fa 04 fa fa fa[04]fa fa fa fd fa
  0x0c047fff8050: fa fa fd fa fa fa fd fd fa fa fd fd fa fa fd fd
  0x0c047fff8060: fa fa fd fd fa fa fd fd fa fa fd fd fa fa fd fd
  0x0c047fff8070: fa fa fd fd fa fa fd fd fa fa fd fd fa fa fd fa
  0x0c047fff8080: fa fa fd fa fa fa fd fa fa fa fd fa fa fa fd fa
  0x0c047fff8090: fa fa fd fa fa fa fd fa fa fa fd fa fa fa fd 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
```

**mdadams** closed this as completed in `41f214b` on Feb 7, 2021

**mdadams** commented on Feb 7, 2021 Collaborator

@dgh05t Thanks for the bug report. This problem is now fixed on the master branch.

**mdadams** mentioned this issue on Feb 7, 2021

jp2_decode() Null Pointer Access #265

Closed

**dgh05t** commented on Feb 9, 2021 Author

CVE-2021-26926 is assigned for this issue, thanks for the efficiency of resolving the issue.

1

**theta682** commented on Mar 1, 2021 Contributor

@mdadams Please release a new version that includes fixes for CVE-2021-26926 and CVE-2021-16927 (#265)

**jubalh** commented on Mar 1, 2021 Member

**@theta682** We did this already more than two weeks ago. Why do you think there is no such version?

Please see the `NEWS` file:

```
2.0.25 (2021-02-07)
===================

* Fix memory-related bugs in the JPEG-2000 codec resulting from
  attempting to decode invalid code streams. (#264, #265)
  This fix is associated with CVE-2021-26926 and CVE-2021-26927.
* Fix wrong return value under some compilers (#260)
* Fix CVE-2021-3272 heap buffer overflow in jp2_decode (#259)
```

**theta682** commented on Mar 1, 2021 Contributor

**@jubalh** Sorry. I just got a notification from NVD that there are new vulnerabilities. However, they are included in 2.0.25.

**thoger** mentioned this issue on Mar 22, 2021

**A null pointer dereference in jp2_decode in jp2_dec.c** #269

Closed

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

4 participants