# Heap-based buffer over-read in get_rgb_row() in rdppm.c #433

Closed sanjeevk001 opened this issue on May 25, 2020 · 3 comments

Assignees

Labels bug fixed

sanjeevk001 commented on May 25, 2020

**Have you searched the existing issues (both open and closed) in the libjpeg-turbo issue tracker to ensure that this bug report is not a duplicate?**
Yes

**Does this bug report describe one of the two known and unsolvable issues with the JPEG format?**
No

**Clear and concise description of the bug:**
Heap-based buffer over-read in get_rgb_row() in rdppm.c

**Steps to reproduce the bug (using *only* libjpeg-turbo):**
Compile with Address Sanitizer (ASan) :
./cjpeg ./reproducer

Without ASan:
valgrind -q ./cjpeg ./reproducer

**Image(s) needed in order to reproduce the bug (if applicable):**

reproducer.zip

**Expected behavior:**

**Observed behavior:**

```
==2127==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x62900000417f at pc 0x55f48a780991 bp 0x7ffccfe84010 sp 0x7ffccfe84000
READ of size 1 at 0x62900000417f thread T0
    #0 0x55f48a780990 in get_rgb_row libjpeg-turbo/rdppm.c:434
    #1 0x55f48a77cadd in main libjpeg-turbo/cjpeg.c:664
    #2 0x7f9476b50b96 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21b96)
    #3 0x55f48a77d049 in _start (libjpeg-turbo/build/cjpeg+0x6049)

0x62900000417f is located 104 bytes to the right of 16151-byte region [0x629000000200,0x629000004117)
allocated by thread T0 here:
    #0 0x7f947730db40 in __interceptor_malloc (/usr/lib/x86_64-linux-gnu/libasan.so.4+0xdeb40)
    #1 0x7f9476fea6cb in alloc_small libjpeg-turbo/jmemmgr.c:318
    #2 0x55f48a785157 in jinit_read_ppm libjpeg-turbo/rdppm.c:756
    #3 0x55f48a77c8cb in select_file_type libjpeg-turbo/cjpeg.c:118
    #4 0x55f48a77c8cb in main libjpeg-turbo/cjpeg.c:636
    #5 0x7f9476b50b96 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21b96)

SUMMARY: AddressSanitizer: heap-buffer-overflow libjpeg-turbo/rdppm.c:434 in get_rgb_row
Shadow bytes around the buggy address:
  0x0c527fff87d0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c527fff87e0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c527fff87f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c527fff8800: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c527fff8810: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c527fff8820: 00 00 07 fa fa fa fa fa fa fa fa fa fa fa fa fa[fa]
  0x0c527fff8830: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c527fff8840: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c527fff8850: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c527fff8860: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c527fff8870: 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
==2127==ABORTING
```

**Platform(s) (compiler version, operating system version, CPU) on which the bug was observed:**

gcc (Ubuntu 7.5.0-3ubuntu1~18.04) 7.5.0, Linux 5.3.0-51-generic

**libjpeg-turbo release(s), commit(s), or branch(es) in which the bug was observed (always test the tip of the master branch or the latest stable pre-release to verify that the bug hasn't already been fixed):**

libjpeg-turbo version 2.0.5 (master)

**If the bug is a regression, the specific commit that introduced the regression (use `git bisect` to determine this):**

**Additional information:**

**sanjeevk001** added the bug label on May 25, 2020

**sanjeevk001** assigned **dcommander** on May 25, 2020

**dcommander** closed this as completed in `3de15e0` on Jun 2, 2020

---

**dcommander** added the fixed label on Jun 2, 2020

**carnil** commented on Jun 3, 2020

This issue got CVE-2020-13790 assigned.

1

**dcommander** commented on Jun 3, 2020 Member

Added CVE ID to the change log. Thanks.

**dcommander** added a commit that referenced this issue on Jun 3, 2020

`rdppm.c: Fix buf overrun caused by bad binary PPM` ··· `5fad352`

**dcommander** added a commit that referenced this issue on Jun 3, 2020

`rdppm.c: Fix buf overrun caused by bad binary PPM` ··· `1f7a5b8`

**dcommander** added a commit that referenced this issue on Jun 3, 2020

`rdppm.c: Fix buf overrun caused by bad binary PPM` ··· `1bfb0b5`

**KexyBiscuit** mentioned this issue on Jun 10, 2020

**libjpeg-turbo: CVE-2020-13790** AOSC-Dev/aosc-os-abbs#2190

Closed

3 tasks

**rain6851** commented on Nov 3, 2020

> This issue got CVE-2020-13790 assigned.
> **@carnil** Can you tell me where did you apply for this CVE? The application I submitted to the CVE website has not yet been replied.

Assignees

**dcommander**

Labels

bug fixed

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

4 participants