GIFLIB Bugs

A library and utilities for processing GIFs
Brought to you by: abadger1999, esr

#159 A heap-buffer-overflow in GIFLIB5.2.1 DumpScreen2RGB() in gif2rgb.c:298:45

Milestone: v1.0_(example) **Status:** open **Owner:** nobody **Labels:** None
Priority: 1
Updated: 2022-08-25 **Created:** 2022-03-29 **Creator:** verf1sh **Private:** No

Environment
- Tested on Ubuntu 20.04.3 LTS x86_64, AFL++
- gcc version 10.3.0
- gif2rgb(5.2.1)

You can reproduce this bug by the follow step:

```
AFL_USE_ASAN=1 make
./gif2rgb giflib_poc
```

AddressSanitizer output

```
==4023907==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x6020000001e0 at pc 0x0000003088e2 bp 0x7ffec2b59590 sp 0x7ffec2b59588
READ of size 1 at 0x6020000001e0 thread T0
#0 0x3088e1 in DumpScreen2RGB /root/gif2rgb.c:298:45
#1 0x3088e1 in GIF2RGB /root/gif2rgb.c:482:5
#2 0x3088e1 in main /root/gif2rgb.c:533:2
#3 0x7f06ece110b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)
#4 0x25182d in _start (/root/gif2rgb+0x25182d)

0x6020000001e0 is located 420 bytes to the right of 12-byte region [0x602000000030,0x60200000003c)
allocated by thread T0 here:
#0 0x2cc862 in calloc (/root/gif2rgb+0x2cc862)
#1 0x31ff29 in GifMakeMapObject /root/gifalloc.c:58:38

SUMMARY: AddressSanitizer: heap-buffer-overflow /root/gif2rgb.c:298:45 in DumpScreen2RGB
Shadow bytes around the buggy address:
0x0c047fff7fe0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff7ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c047fff8000: fa fa 03 fa fa fa 00 04 fa fa fa fa fa fa fa fa
0x0c047fff8010: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8020: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
=>0x0c047fff8030: fa fa fa fa fa fa fa fa fa fa fa fa[fa]fa fa fa
0x0c047fff8040: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8050: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8060: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8070: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c047fff8080: 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
==4023907==ABORTING
```

Thank you.

1 Attachments

giflib_poc.zip

Discussion

Rajat Aggarwal - *2022-05-30*

The reported CVE is of high severity rated as 8.8 on NVD https://nvd.nist.gov/vuln/detail/CVE-2022-28506#vulnCurrentDescriptionTitle.
I would really appreciate any information about the approximate timeline for fixing this vulnerability either through a patch or by the next official release.

Thanks
Rajat

Eric S. Raymond - *2022-06-08*

Rajat Aggarwal rajatnituk@users.sourceforge.net:

> The reported CVE is of high severity rated as 8.8 on NVD
> https://nvd.nist.gov/vuln/detail/CVE-2022-28506#vulnCurrentDescriptionTitle.
> I would really appreciate any information about the approximate timeline for fixing this
> vulnerability either through a patch or by the next official release.

I've traveling to a conference on Thurdsday. I may have time to fix it
tomorrow; if not, early next week.
--
Eric S. Raymond

Matej Mužila - *2022-05-30*

I created a fix for this issue: https://sourceforge.net/p/giflib/code/merge-requests/11/

Todd Wohlers - *2022-08-11*

Is this bug, #159 (CVS-2022-28506), a duplicate of #151 (CVE-2020-23922)?

Rajat Aggarwal - *2022-08-25*

I saw that the merge request for patch is still in pending state. Any info on solving this issue either by this patch/or some other patch would be really helpful.

Thanks,
Rajat

Log in to post a comment.