

New issue

Jump to bottom

A heap overflow in bits.c:1424 #261

Closed

seviezhou opened this issue on Aug 2, 2020 · 1 comment

Assignees



Labels

bug

fuzzing

Milestone

0.11

seviezhou commented on Aug 2, 2020

System info

Ubuntu X64, gcc (Ubuntu 5.5.0-12ubuntu1), dwg2dxf (latest master 39ef943)

Configure

CONFIGURE_FLAGS="-g -fsanitize=address" LD_FLAGS="-fsanitize=address" ./configure

Command line

./programs/dwg2dxf -b -m ./SEGV-check_POLYLINE_handles-decode-5110 -o /dev/null

AddressSanitizer output

```
=====
==65289==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x62c0000381ff at pc 0x7fc1be68a945 bp 0x7fff3419f990 sp 0x7fff3419f138
READ of size 6 at 0x62c0000381ff thread T0
#0 0x7fc1be68a944 in __asan_memcpy (/usr/lib/x86_64-linux-gnu/libasan.so.2+0x8c944)
#1 0x5588916f116f in memcpy /usr/include/x86_64-linux-gnu/bits/string_fortified.h:34
#2 0x5588916f116f in bit_read_fixed /home/seviezhou/libredwg/src/bits.c:1424
#3 0x558891715678 in acds_private /home/seviezhou/libredwg/src/acds.spec:111
#4 0x5588917b3161 in read_2004_section_acds /home/seviezhou/libredwg/src/decode.c:3437
#5 0x5588917b3161 in decode_R2004 /home/seviezhou/libredwg/src/decode.c:3694
#6 0x5588917bf646 in dwg_decode /home/seviezhou/libredwg/src/decode.c:242
#7 0x5588916b89fc in dwg_read_file /home/seviezhou/libredwg/src/dwg.c:251
#8 0x5588916b5e12 in main /home/seviezhou/libredwg/programs/dwg2dxf.c:258
#9 0x7fc1bde90b96 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21b96)
#10 0x5588916b6d69 in _start (/home/seviezhou/libredwg/programs/dwg2dxf+0xa88d69)



0x62c0000381ff is located 1 bytes to the left of 29696-byte region [0x62c000038200,0x62c00003f600)
allocated by thread T0 here:
#0 0x7fc1be6967aa in __interceptor_malloc (/usr/lib/x86_64-linux-gnu/libasan.so.2+0x987aa)
#1 0x5588917211c4 in read_2004_compressed_section /home/seviezhou/libredwg/src/decode.c:2432
#2 0x5588922c13aa (/home/seviezhou/libredwg/programs/dwg2dxf+0x16933aa)

SUMMARY: AddressSanitizer: heap-buffer-overflow ??:0 __asan_memcpy
Shadow bytes around the buggy address:
0x0c587ffffe00: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c587ffffe08: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c587ffffe10: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c587ffffe18: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c587ffffe20: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
=>0x0c587ffffe30: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa[fa]
0x0c587ffffe40: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c587ffffe50: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c587ffffe60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c587ffffe70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c587ffffe80: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
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
Heap right redzone: fb
Freed heap region: fd
Stack left redzone: f1
Stack mid redzone: f2
Stack right redzone: f3
Stack partial redzone: f4
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
==65289==ABORTING
```

POC

heap-overflow-bit_read_fixed-bits-1424.zip

  **rurban** self-assigned this on Aug 2, 2020

  **rurban** added `bug` `fuzzing` labels on Aug 2, 2020

  **rurban** added this to the **0.11** milestone on Aug 2, 2020

rurban commented on Aug 2, 2020

Contributor

This was already fixed with the ACDS.num_segidxf check in GH [#259](#)
Not repro anymore

 **rurban** closed this as completed on Aug 2, 2020

Assignees

 **rurban**

Labels

`bug` `fuzzing`

Projects

None yet

Milestone

0.11

Development

No branches or pull requests

2 participants