New issue

Jump to bottom

Stack buffer overflow in function dimC_box_read at isomedia/box_code_3gpp.c:1070 #2296

Closed Janette88 opened this issue on Oct 30 · 1 comment

Janette88 commented on Oct 30 • edited

Description

Stack buffer overflow in function dimC_box_read at isomedia/box_code_3gpp.c:1070

System info

ubuntu 20.04 lts

version info:

...
/MP4Box -version
MP4Box - GPAC version 2.1-DEV-rev428-gcb8ae46c8-master
(c) 2000-2022 Telecom Paris distributed under LGPL v2.1+ - http://gpac.io

Please cite our work in your research:
GPAC Filters: https://doi.org/10.1145/3339825.3394929
GPAC: https://doi.org/10.1145/1291233.1291452

GPAC Configuration: --enable-sanitizer
Features: GPAC_CONFIG_LINUX GPAC_64_BITS GPAC_HAS_IPV6 GPAC_HAS_SOCK_UN GPAC_MINIMAL_ODF GPAC_HAS_QJS GPAC_HAS_LINUX_DVB
...

compile

./configure --enable-sanitizer
make

crash command:
./MP4Box -bt poc2

poc2 :
poc2.zip

Here is stack overflow output by ASAN:

```
[AV1] Error parsing tile group, tile 0 start 58 + size 17220 exceeds OBU length 3
[AV1] Frame parsing did not consume the right number of bytes !
[AV1] could not parse AV1 OBU at position 42. Leaving parsing.
[ISOBMFF] AV1ConfigurationBox overflow read 17 bytes, of box size 16.
[iso file] Box "av1C" size 24 (start 20) invalid (read 25)
=================================================================
==22786==ERROR: AddressSanitizer: stack-buffer-overflow on address 0x7fff0c1f8a40 at pc 0x7f7bb77cb3ad bp 0x7fff0c1f85d0 sp 0x7fff0c1f7d78
READ of size 1031 at 0x7fff0c1f8a40 thread T0
    #0 0x7f7bb77cb3ac in __interceptor_strdup ../../../../src/libsanitizer/asan/asan_interceptors.cc:443
    #1 0x7f7bb43ee2dd in dimC_box_read isomedia/box_code_3gpp.c:1070
    #2 0x7f7bb44aca33 in gf_isom_box_read isomedia/box_funcs.c:1866
    #3 0x7f7bb44aca33 in gf_isom_box_parse_ex isomedia/box_funcs.c:271
    #4 0x7f7bb44ade85 in gf_isom_parse_root_box isomedia/box_funcs.c:38
    #5 0x7f7bb44d6efc in gf_isom_parse_movie_boxes_internal isomedia/isom_intern.c:378
    #6 0x7f7bb44dd111 in gf_isom_parse_movie_boxes isomedia/isom_intern.c:868
    #7 0x7f7bb44dd111 in gf_isom_open_file isomedia/isom_intern.c:988
    #8 0x55829fb43139 in mp4box_main /home/fuzz/gpac/applications/mp4box/mp4box.c:6211
    #9 0x7f7bb1a59082 in __libc_start_main ../csu/libc-start.c:308
    #10 0x55829fb1ecbd in _start (/home/fuzz/gpac/bin/gcc/MP4Box+0xa3cbd)

Address 0x7fff0c1f8a40 is located in stack of thread T0 at offset 1056 in frame
    #0 0x7f7bb43edeff in dimC_box_read isomedia/box_code_3gpp.c:1048

  This frame has 1 object(s):
    [32, 1056) 'str' (line 1049) <== Memory access at offset 1056 overflows this variable
HINT: this may be a false positive if your program uses some custom stack unwind mechanism, swapcontext or vfork
      (longjmp and C++ exceptions *are* supported)
SUMMARY: AddressSanitizer: stack-buffer-overflow ../../../../src/libsanitizer/asan/asan_interceptors.cc:443 in __interceptor_strdup
Shadow bytes around the buggy address:
  0x1000618370f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x100061837100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x100061837110: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x100061837120: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x100061837130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x100061837140: 00 00 00 00 00 00 00 00[f3]f3 f3 f3 f3 f3 f3 f3
  0x100061837150: f3 f3 f3 f3 f3 f3 f3 f3 00 00 00 00 00 00 00 00
  0x100061837160: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x100061837170: f1 f1 f1 f1 f1 f1 f1 f8 f2 00 f2 f2 f2 00 00 f3 f3
  0x100061837180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x100061837190: 00 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
  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
==22786==ABORTING
```

Impact

This is capable of causing crashes and allowing modification of stack memory which could lead to remote code execution.

Code location

```
GF_Err dimC_box_read(GF_Box *s, GF_BitStream *bs)
{
        char str[1024];
        u32 i;
        GF_DIMSSceneConfigBox *p = (GF_DIMSSceneConfigBox *)s;

        ISOM_DECREASE_SIZE(p, 3);
        p->profile = gf_bs_read_u8(bs);
        p->level = gf_bs_read_u8(bs);
        p->pathComponents = gf_bs_read_int(bs, 4);
        p->fullRequestHost = gf_bs_read_int(bs, 1);
        p->streamType = gf_bs_read_int(bs, 1);
        p->containsRedundant = gf_bs_read_int(bs, 2);

        i=0;
        str[0]=0;
        while (i < GF_ARRAY_LENGTH(str)) {
                str[i] = gf_bs_read_u8(bs);
                if (!str[i]) break;
                i++;
        }
        ISOM_DECREASE_SIZE(p, i);

        **p->textEncoding = gf_strdup(str);**           //line:1070   this issue

        i=0;
        str[0]=0;
        while (i < GF_ARRAY_LENGTH(str)) {
                str[i] = gf_bs_read_u8(bs);
                if (!str[i]) break;
                i++;
        }
        ISOM_DECREASE_SIZE(p, i);

        p->contentEncoding = gf_strdup(str);                        //line:1081   issue 2294 related
        return GF_OK;
}
```

jeanlf commented 29 days ago — Contributor

fixed by fixing #2294 thanks for the poc

jeanlf closed this as completed 29 days ago

Janette88 mentioned this issue 26 days ago

Memory Leak in dimC_box_read at isomedia/box_code_3gpp.c:1060 #2307

Closed

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants