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

Closed Janette88 opened this issue 26 days ago · 1 comment

Janette88 commented 26 days ago • edited

Description

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

System info

ubuntu 20.04 lts

version info:

```
./MP4Box -version
MP4Box - GPAC version 2.1-DEV-rev460-g9d963dc62-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:
Features: GPAC_CONFIG_LINUX GPAC_64_BITS GPAC_HAS_IPV6 GPAC_HAS_SOCK_UN GPAC_MINIMAL_ODF GPAC_HAS_QJS GPAC_HAS_LINUX_DVB  GPAC_DI
```

compile

./configure --enable-sanitizer
make

crash command:

./MP4Box -bt poc_ml

poc :
https://github.com/Janette88/test_pocs/blob/main/poc_ml

Here is output by ASAN:

```
[ISOBMFF] AV1ConfigurationBox: read only 4 bytes (expected 16).
[iso file] Box "av1C" (start 20) has 12 extra bytes
[isom] not enough bytes in box dimC: 0 left, reading 1 (file isomedia/box_code_3gpp.c, line 1082)
[iso file] Read Box "dimC" (start 44) failed (Invalid IsoMedia File) - skipping
Error opening file /home/fuzz/test/poc_ml: Invalid IsoMedia File

=================================================================
==71539==ERROR: LeakSanitizer: detected memory leaks

Direct leak of 2566 byte(s) in 1 object(s) allocated from:
    #0 0x7fe8c635f808 in __interceptor_malloc ../../../../src/libsanitizer/asan/asan_malloc_linux.cc:144
    #1 0x7fe8c2ef8d39 in dimC_box_read isomedia/box_code_3gpp.c:1060
    #2 0x7fe8c2fb75c3 in gf_isom_box_read isomedia/box_funcs.c:1866
    #3 0x7fe8c2fb75c3 in gf_isom_box_parse_ex isomedia/box_funcs.c:271
    #4 0x7fe8c2fb8a15 in gf_isom_parse_root_box isomedia/box_funcs.c:38
    #5 0x7fe8c2fe1a8c in gf_isom_parse_movie_boxes_internal isomedia/isom_intern.c:378
    #6 0x7fe8c2fe7ca1 in gf_isom_parse_movie_boxes isomedia/isom_intern.c:868
    #7 0x7fe8c2fe7ca1 in gf_isom_open_file isomedia/isom_intern.c:988
    #8 0x55c56a3e9139 in mp4box_main /home/fuzz/gpac/applications/mp4box/mp4box.c:6209
    #9 0x7fe8c0558082 in __libc_start_main ../csu/libc-start.c:308

SUMMARY: AddressSanitizer: 2566 byte(s) leaked in 1 allocation(s).
```

code location:

```
GF_Err dimC_box_read(GF_Box *s, GF_BitStream *bs)
{
        u32 i, msize;
        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);

        char *str = gf_malloc(sizeof(char)*(p->size+1));       //line 1060   here p->size+1 = 2566
        if (!str) return GF_OUT_OF_MEM;
        msize = (u32) p->size;
        str[msize] = 0;
        i=0;
        str[0]=0;
        while (i < msize) {
                ISOM_DECREASE_SIZE(p, 1);
                str[i] = gf_bs_read_u8(bs);
                if (!str[i]) break;
```

```
                  i++;
          }
          if (i == msize) {
                  gf_free(str);
                  return GF_ISOM_INVALID_FILE;
          }

          p->textEncoding = gf_strdup(str);

          i=0;
          str[0]=0;
          while (i < msize) {
                  ISOM_DECREASE_SIZE(p, 1);            //line :1082 not enough bytes in box dimC: 0 left, reading 1

                  str[i] = gf_bs_read_u8(bs);
                  if (!str[i]) break;
                  i++;
          }
          if (i == msize) {
                  gf_free(str);
                  return GF_ISOM_INVALID_FILE;
          }

          p->contentEncoding = gf_strdup(str);
          gf_free(str);
          if (!p->textEncoding || !p->contentEncoding)
                  return GF_OUT_OF_MEM;
          return GF_OK;
  }
```

ps: The issue could be verified using the poc in issue 2294 and 2296. The patch of issue 2294 and 2296 was not perfect because it still existed memory leak risk .

ref:
#2294
#2296

jeanlf closed this as completed in f045be5 26 days ago

jeanlf commented 26 days ago Contributor

indeed thanks for the cross-check

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants