gpac / **gpac** Public

<> Code  ⊙ **Issues** 11  Pull requests 2  Wiki  Security  Insights

New issue

Jump to bottom

# NULL Pointer Dereference still exists in gf_isom_parse_movie_boxes_internal #2163

Closed **0xdd96** opened this issue on Apr 1 · 1 comment

**0xdd96** commented on Apr 1

version info:

```
root@d8a714203f6e:# ./MP4Box -version
MP4Box - GPAC version 2.1-DEV-rev87-g053aae8-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: --prefix=/path_to_gpac/build --enable-debug --enable-sanitizer
Features: GPAC_CONFIG_LINUX GPAC_64_BITS GPAC_HAS_IPV6 GPAC_HAS_SSL GPAC_HAS_SOCK_UN
GPAC_MINIMAL_ODF GPAC_HAS_QJS GPAC_HAS_FAAD GPAC_HAS_MAD GPAC_HAS_LIBA52 GPAC_HAS_JPEG
GPAC_HAS_PNG GPAC_HAS_FFMPEG GPAC_HAS_JP2 GPAC_HAS_THEORA GPAC_HAS_VORBIS GPAC_HAS_XVID
GPAC_HAS_LINUX_DVB
```

**poc:**poc
**command:** MP4Box -hint -out /dev/null $poc$
**crash:**

```
root@d8a714203f6e:# ./MP4Box -hint -out /dev/null poc
[iso file] Read Box type 00000000 (0x00000000) at position 45 has size 0 but is not at root/file
level. Forbidden, skipping end of parent box !
[iso file] Read Box "abst" (start 0) failed (Unknown Error (10)) - skipping
isomedia/isom_intern.c:392:12: runtime error: member access within null pointer of type 'struct
GF_Box'
```

When `size=0` and `is_root_box=false`, `gf_isom_box_parse_ex` will return `GF_SKIP_BOX` (i.e., 10) at line 138 of box_funcs.c.

gpac/src/isomedia/box_funcs.c

Lines 129 to 142 in `7f060bb`

```
129     if (!size) {
130             if (is_root_box) {
131                     if (!skip_logs) {
132                             GF_LOG(GF_LOG_DEBUG, GF_LOG_CONTAINER, ("[iso file] Warning Rea
133                     }
134                     size = gf_bs_available(bs) + 8;
135             } else {
136                     if (!skip_logs) {
137                             GF_LOG(GF_LOG_ERROR, GF_LOG_CONTAINER, ("[iso file] Read Box ty
138                             return GF_SKIP_BOX;
```

This will cause `*outBox` to be set to NULL (in box_funcs.c:312) and the return value `GF_SKIP_BOX` will be passed to the upper function ( in box_funcs.c:318).

gpac/src/isomedia/box_funcs.c
Lines 310 to 319 in `7f060bb`

```
310     if (e && (e != GF_ISOM_INCOMPLETE_FILE)) {
311             gf_isom_box_del(newBox);
312             *outBox = NULL;
313
314             if (!skip_logs) {
315                     GF_LOG(GF_LOG_ERROR, GF_LOG_CONTAINER, ("[iso file] Read Box \"%s\" (start "
316             }
317             //we don't try to reparse known boxes that have been failing (too dangerous)
318             return e;
319     }
```

The program now executes the empty if block when `e>=0` ( in isom_intern.c:375-377), and later dereferences the null pointer in line 392 of isom_intern.c.

gpac/src/isomedia/isom_intern.c
Lines 373 to 392 in `7f060bb`

```
373     e = gf_isom_parse_root_box(&a, mov->movieFileMap->bs, boxType, bytesMissing, progressi
374
375     if (e >= 0) {
376
377     } else if (e == GF_ISOM_INCOMPLETE_FILE) {
378             /*our mdat is uncomplete, only valid for READ ONLY files...*/
379             if (mov->openMode != GF_ISOM_OPEN_READ) {
380                     GF_LOG(GF_LOG_ERROR, GF_LOG_CONTAINER, ("[iso file] Incomplete MDAT whi
381                     return GF_ISOM_INVALID_FILE;
382             }
```

Note that although the crash path is the same as in issue #2155, their root cause is different.

👍 1

**hmnthabit** commented on Apr 1

Thanks for reporting this

What's the problem of checking `a` strucut before the switch statement? **@jeanlf**

Also, it's better to test the poc against all the arguments that do file parsing.

**jeanlf** closed this as completed in `37592ad` on Apr 12

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants