New issue

Jump to bottom

GPAC-2.0.0 MP4Box: stack overflow with unlimited length and controllable content in diST_box_read #2175

Closed **xidoo123** opened this issue on Apr 16 · 1 comment

xidoo123 commented on Apr 16 • edited

Description

When GPAC tries to parse a MP4 file, it calls the function `diST_box_read()` to read from video. In this funtion, it allocates a buffer `str` with fixed length. However, content read from `bs` is controllable by user, so is the length, which causes a buffer overflow.

```
char str[1024];

i=0;
str[0]=0;
while (1) {
    str[i] = gf_bs_read_u8(bs);
    if (!str[i]) break;
    i++;
}
```

Impact

Since video content is absolutely controllable by users, an unlimited length will cause stack overflow, corrupting canary or even get shell.

Mitigation

We can just set a length limit to it, making it less than 1024 byte. See pull request #2174 .

Reproduce

On Ubuntu 2004, make with this.

```
./configure --static-bin
make
```

Run the following command with POC.mp4.

```
$ MP4Box -info ./POC.mp4
```

You may get a stack smashing detectde error, which indicates that CANARY is crashed.

```
[BS] Attempt to overread bitstream
*** stack smashing detected ***: terminated
Aborted
```

GDB

```
*** stack smashing detected ***: terminated

Program received signal SIGABRT, Aborted.
0x0000000000aa31eb in raise ()

pwndbg> bt
#0  diST_box_read (s=0xdf4b00, bs=0xdf71e0) at isomedia/box_code_3gpp.c:1130
#1  0x000000000052e8c9 in gf_isom_box_read (bs=0xdf71e0, a=0xdf4b00) at isomedia/box_funcs.c:1832
#2  gf_isom_box_parse_ex (outBox=outBox@entry=0x7fffffff8540, bs=<optimized out>,
bs@entry=0xdf71e0, parent_type=parent_type@entry=0, is_root_box=is_root_box@entry=GF_TRUE) at
isomedia/box_funcs.c:264
#3  0x000000000052f070 in gf_isom_parse_root_box (outBox=outBox@entry=0x7fffffff8540, bs=0xdf71e0,
box_type=box_type@entry=0x0, bytesExpected=bytesExpected@entry=0x7fffffff8590,
progressive_mode=progressive_mode@entry=GF_FALSE) at isomedia/box_funcs.c:38
#4  0x0000000000536af8 in gf_isom_parse_movie_boxes_internal (mov=mov@entry=0xdf6fc0,
boxType=boxType@entry=0x0, bytesMissing=bytesMissing@entry=0x7fffffff8590,
progressive_mode=progressive_mode@entry=GF_FALSE) at isomedia/isom_intern.c:373
#5  0x0000000000538287 in gf_isom_parse_movie_boxes (progressive_mode=GF_FALSE,
bytesMissing=0x7fffffff8590, boxType=0x0, mov=0xdf6fc0) at isomedia/isom_intern.c:852
#6  gf_isom_open_file (fileName=0x7fffffffe67d "../../../../crashes/1.mp4", OpenMode=<optimized
out>, tmp_dir=0x0) at isomedia/isom_intern.c:972
#7  0x0000000000414dd4 in mp4boxMain (argc=<optimized out>, argv=<optimized out>) at main.c:5968
#8  0x0000000000a94000 in __libc_start_main ()
#9  0x0000000000402e6e in _start () at main.c:6585
```

Credits

xdchase

POC

POC.zip

 **jeanlf** closed this as completed in `3dbe11b` on Apr 19

  **jeanlf** mentioned this issue on Apr 19

add length limit in diST_box_read() #2174

Closed

pedrohc commented on Apr 22

CVE-2022-1441 was assigned to this issue.

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants