New issue

Jump to bottom

Heap-buffer-overflow isomedia/stbl_read.c:135 in stbl_GetSampleSize() #1482

Closed **14isnot40** opened this issue on May 12, 2020 · 3 comments

14isnot40 commented on May 12, 2020

- [y] I looked for a similar issue and couldn't find any.
- [y] I tried with the latest version of GPAC. Installers available at http://gpac.io/downloads/gpac-nightly-builds/
- [y] I give enough information for contributors to reproduce my issue (meaningful title, github labels, platform and compiler, command-line ...). I can share files anonymously with this dropbox: https://www.mediafire.com/filedrop/filedrop_hosted.php?drop=eec9e058a9486fe4e99c33021481d9e1826ca9dbc242a6cfaab0fe95da5e5d95

Describe the bug
A heap-based buffer overflow was discovered in libgpac, during structure GF_SampleSizeBox 'stsz' member 'sizes' points to an invalid address. The issue is being triggered in the function stbl_GetSampleSize() at isomedia/stbl_read.c

To Reproduce
Steps to reproduce the behavior:

1. Compile according to the default configuration

```
$ CC="gcc -fsanitize=address -g" CXX="g++ -fsanitize=address -g" ./configure --static-mp4box
$ make
```

2. execute command

```
MP4Box -hint $poc
```

poc can be found here.

Expected behavior
An attacker can exploit this vulnerability by submitting a malicious media file that exploits this issue. This will result in a Denial of Service (DoS) and potentially Information Exposure when the application attempts to process the file.

Screenshots
ASAN Reports

```
==94786==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x61400000fdd0 at pc 0x000000744231 bp 0x7fffffff83c0 sp 0x7fffffff83b0
READ of size 4 at 0x61400000fdd0 thread T0
    #0 0x744230 in stbl_GetSampleSize isomedia/stbl_read.c:135
    #1 0x717f3d in Media_GetSample isomedia/media.c:418
    #2 0x6cd966 in gf_isom_get_sample_info isomedia/isom_read.c:1692
    #3 0x912ed8 in gf_media_get_sample_average_infos media_tools/isom_hinter.c:54
    #4 0x913d43 in gf_hinter_track_new media_tools/isom_hinter.c:560
    #5 0x41e02e in HintFile (/usr/local/bin/MP4Box+0x41e02e)
    #6 0x429806 in mp4boxMain (/usr/local/bin/MP4Box+0x429806)
    #7 0x7ffff615e82f in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x2082f)
    #8 0x41d668 in _start (/usr/local/bin/MP4Box+0x41d668)

0x61400000fdd0 is located 0 bytes to the right of 400-byte region [0x61400000fc40,0x61400000fdd0)
allocated by thread T0 here:
    #0 0x7ffff6f02961 in realloc (/usr/lib/x86_64-linux-gnu/libasan.so.2+0x98961)
    #1 0x7516dd in stbl_AppendSize isomedia/stbl_write.c:1487

SUMMARY: AddressSanitizer: heap-buffer-overflow isomedia/stbl_read.c:135 stbl_GetSampleSize
Shadow bytes around the buggy address:
  0x0c287fff9f60: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c287fff9f70: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c287fff9f80: fa fa fa fa fa fa fa fa 00 00 00 00 00 00 00 00
  0x0c287fff9f90: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c287fff9fa0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c287fff9fb0: 00 00 00 00 00 00 00 00 00 00[fa]fa fa fa fa fa
  0x0c287fff9fc0: fa fa fa fa fa fa fa fa fd fd fd fd fd fd fd fd
  0x0c287fff9fd0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c287fff9fe0: fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd fd
  0x0c287fff9ff0: fd fd fd fd fd fd fd fd fd fd fd fd fa fa fa fa
  0x0c287fffa000: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
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
==94786==ABORTING
```

Possible causes of vulnerabilities
structure GF_SampleSizeBox 'stsz' member 'sizes' points to an invalid address

```
GF_Err stbl_GetSampleSize(GF_SampleSizeBox *stsz, u32 SampleNumber, u32 *Size)
{
	if (!stsz || !SampleNumber || SampleNumber > stsz->sampleCount) return GF_BAD_PARAM;

	(*Size) = 0;

	if (stsz->sampleSize && (stsz->type != GF_ISOM_BOX_TYPE_STZ2)) {
		(*Size) = stsz->sampleSize;
	} else if (stsz->sizes) {
		(*Size) = stsz->sizes[SampleNumber - 1];
	}
	return GF_OK;
}
```

System (please complete the following information):

- OS version : Ubuntu 16.04
- GPAC Version : GPAC 0.8.0-e10d39d-master branch

jeanlf commented on Jun 11, 2020 — Contributor

fixed, thanks for the report

jeanlf closed this as completed on Jun 11, 2020

carnil commented on Sep 27, 2021

Bisecting this issue seems to indicate that it is fixed with `e4ed32b` , but this does not match the closing date on Jun 11 2020. Does it still make sense?

carnil commented on Sep 27, 2021

```
git bisect log
git bisect start '--term-old' 'broken' '--term-new' 'fixed'
# broken: [e10d39d93062d83c248354762f81c5fc58b51e2a] added missing signaling of PASP in avc import
git bisect broken e10d39d93062d83c248354762f81c5fc58b51e2a
# fixed: [a99c032b7afbc2e0a55d0259fd09b4139f8a7402] updated gitignore for dep files - [noCI]
git bisect fixed a99c032b7afbc2e0a55d0259fd09b4139f8a7402
# broken: [7f5985d701df1ab081c987948cffeb83b8e81c65] added advanced ttxt test
git bisect broken 7f5985d701df1ab081c987948cffeb83b8e81c65
# fixed: [a99c032b7afbc2e0a55d0259fd09b4139f8a7402] updated gitignore for dep files - [noCI]
git bisect fixed a99c032b7afbc2e0a55d0259fd09b4139f8a7402
# skip: [d973b7e044c59a7a06d9d129665ea7190122fedc] updated QuickJS modification list [noCI]
git bisect skip d973b7e044c59a7a06d9d129665ea7190122fedc
# fixed: [b75ae95df5261541cf2813fd8de181798d1d54a9] Fixed compil with 3D disabled
git bisect fixed b75ae95df5261541cf2813fd8de181798d1d54a9
# broken: [9d778e97825c82593fbf2cb590663625be436c3f] fixed bug in wave formating
git bisect broken 9d778e97825c82593fbf2cb590663625be436c3f
# broken: [2676e067cf8cf289f24702ebe1ece4f78f937778] fixed property copy bug in dash demux
git bisect broken 2676e067cf8cf289f24702ebe1ece4f78f937778
# broken: [8251cdfba5b68539b12dce05cbc5831f41c344df] updated doc for atscin
git bisect broken 8251cdfba5b68539b12dce05cbc5831f41c344df
# fixed: [df0c205e63f22c125845340e4ddc66b97e32c726] fixed broken next earliest cts compute when mutliple moof in segment
git bisect fixed df0c205e63f22c125845340e4ddc66b97e32c726
# broken: [00e5324c00e86d40fa07d6046ff8eee8da88c5f1] cleanup on 0-length profile
git bisect broken 00e5324c00e86d40fa07d6046ff8eee8da88c5f1
# fixed: [33a7b4b1fa9fd635ea7b1a11199716ffd08971ea] fixed bug in bitstream seek when using block dispatch mode
git bisect fixed 33a7b4b1fa9fd635ea7b1a11199716ffd08971ea
# fixed: [0b63edf9bfdae555907a055062ca9fc471126e38] updated doc [noCI]
git bisect fixed 0b63edf9bfdae555907a055062ca9fc471126e38
# fixed: [a717c795c35f51ac92774d97aa2d1bba1221bc8f] optimized filter packet queue browsing
git bisect fixed a717c795c35f51ac92774d97aa2d1bba1221bc8f
# broken: [442c9f731f6139140bc7dbd92d86995ac6b5d09d] changed log format for -comp
git bisect broken 442c9f731f6139140bc7dbd92d86995ac6b5d09d
# fixed: [e4ed32bf56fc02fb8a04b9e13f4d7bdae2b3ae12] fixed potential crash in traf merging when packed samples are used
git bisect fixed e4ed32bf56fc02fb8a04b9e13f4d7bdae2b3ae12
# first fixed commit: [e4ed32bf56fc02fb8a04b9e13f4d7bdae2b3ae12] fixed potential crash in traf merging when packed samples are used
```

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

3 participants