New issue

Jump to bottom

A integer overflow in function filter_core/filter_props.c:gf_props_assign_value #1718

Closed **treebacker** opened this issue on Mar 28, 2021 · 1 comment

treebacker commented on Mar 28, 2021 • edited

There is a integer overflow in function `filter_core/filter_props.c:gf_props_assign_value` .
In which, the arg `const GF_PropertyValue *value` , maybe value->value.data.size is a negative number.
In result, memcpy in gf_props_assign_value failed.
More, this bug may result a heap overflow with crafted file.

In command line:
./bin/gcc/gpac -info bug.flac

```
ubuntu@VM-0-3-ubuntu:~/[redacted]gpac-1.0.1$ ./bin/gcc/gpac -version
Nothing to do, check usage "gpac -h"
gpac - GPAC command line filter engine - version 1.0.1-revUNKNOWN_REV
(c) 2000-2020 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

ubuntu@VM-0-3-ubuntu:~/[redacted]$ make^C
ubuntu@VM-0-3-ubuntu:~/[redacted]/gpac-1.0.1$ ./bin/gcc/gpac -info [redacted]/r8k_2c_8b.flac
Segmentation fault
```

In gdb:

```
Program received signal SIGSEGV, Segmentation fault.
__memmove_avx_unaligned_erms () at ../sysdeps/x86_64/multiarch/memmove-vec-unaligned-erms.S:435
435     ../sysdeps/x86_64/multiarch/memmove-vec-unaligned-erms.S: No such file or directory.
(gdb) bt
#0  __memmove_avx_unaligned_erms () at ../sysdeps/x86_64/multiarch/memmove-vec-unaligned-erms.S:435
#1  0x00007ffff7793314 in gf_props_assign_value (prop=0x55555579cd90, value=0x7fffffffe270, is_old_prop=GF_FALSE) at filter_core/filter_props.c:809
#2  0x00007ffff779354b in gf_props_insert_property (map=0x555555790220, hash=0, p4cc=1145259591, name=0x0, dyn_name=0x0, value=0x7fffffffe270) at filter_core/filter_props.c
#3  0x00007ffff77935f0 in gf_props_set_property (map=0x555555790220, p4cc=1145259591, name=0x0, dyn_name=0x0, value=0x7fffffffe270) at filter_core/filter_props.c:889
#4  0x00007ffff7788667 in gf_filter_pid_set_property_full (pid=0x5555557a4b10, prop_4cc=1145259591, prop_name=0x0, dyn_name=0x0, value=0x7fffffffe270, is_info=GF_FALSE)
    at filter_core/filter_pid.c:4337
#5  0x00007ffff77886c3 in gf_filter_pid_set_property (pid=0x5555557a4b10, prop_4cc=1145259591, value=0x7fffffffe270) at filter_core/filter_pid.c:4344
#6  0x00007ffff78c7023 in flac_dmx_check_pid (filter=0x5555557a5d70, ctx=0x55555579d280,
    ds=0x5555557acd54 "L_\222\273t?>\337\377]\357\225\071\305V1\211n»x\362\\\362\a\255]\030\033\305\373\220É\033\064\017\217V\217\212C\031\305\025\006\071\342bŸ\253\202\377
\313\305Z\263\344\301\211\221C\321oB\204\325\f\036\021\004rs\205f\376S{\223Jf\2730\264\272\376\330\346\326\360k\322D\262\212\v\t\202\336\377\227\246\240\066s\022\200\300\230
0\372\254\313;qTm$\344\371\261\236Y\001\371+\227`jS\200R\347\320]A'!\247\217\353\211~\346\230zj\264\257\323\022\345\315F9Kd\006\002\245\332\034\371D\023\234\360\001\356b[:w
size=4294967292) at filters/reframe_flac.c:162
#7  0x00007ffff78c812e in flac_dmx_process (filter=0x5555557a5d70) at filters/reframe_flac.c:517
#8  0x00007ffff77a79fc in gf_filter_process_task (task=0x555555790060) at filter_core/filter.c:2158
#9  0x00007ffff7799b91 in gf_fs_thread_proc (sess_thread=0x5555557866c0) at filter_core/filter_session.c:1467
#10 0x00007ffff779a49a in gf_fs_run (fsess=0x555555786630) at filter_core/filter_session.c:1704
#11 0x000055555555e49c in gpac_main (argc=3, argv=0x5555557872b0) at main.c:2116
#12 0x00005555555e771 in main (argc=3, argv=0x7fffffffe878) at main.c:2171
(gdb) b filter_core/filter_props.c:809
Breakpoint 2 at 0x7ffff77932f0: file filter_core/filter_props.c, line 809.
(gdb) r
The program being debugged has been started already.
Start it from the beginning? (y or n) y
Starting program: /home/ubuntu/treebacker/fuzzwork/dataset/tprogram/gpac-1.0.1/bin/gcc/gpac -info ~/treebacker/fuzzwork/input_cve/gpac/r8k_2c_8b.flac
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".

Breakpoint 2, gf_props_assign_value (prop=0x55555579cd90, value=0x7fffffffe270, is_old_prop=GF_FALSE) at filter_core/filter_props.c:809
809             memcpy(prop->prop.value.data.ptr, value->value.data.ptr, value->value.data.size);
(gdb) p value->value.data.size
$1 = 4294967292
(gdb)
```

0xfffffffc

The crafted file is in attach zip:
bug.zip

jeanlf commented on Mar 29, 2021 Contributor

fixed here, thanks for the report

jeanlf closed this as completed on Mar 29, 2021

Assignees
No one assigned

Labels
None yet

Projects
None yet

Milestone
No milestone

Development
No branches or pull requests

2 participants