New issue

Jump to bottom

SEGV in slice.cc #298

Closed **dhbbb** opened this issue on Jun 22, 2021 · 4 comments

dhbbb commented on Jun 22, 2021 • edited

Hello,
A SEGV has occurred when running program dec265,
System info:
Ubuntu 20.04.1 : clang 10.0.0 , gcc 9.3.0

Dec265 v1.0.8

poc (1).zip

Verification steps:
1.Get the source code of libde265
2.Compile

```
cd libde265
mkdir build && cd build
cmake ../ -DCMAKE_CXX_COMPILER=clang++ -DCMAKE_CXX_FLAGS="fsanitize=address"
make -j 32
```

3.run dec265(without asan)

```
./dec265 poc
```

Output

```
WARNING: end_of_sub_stream_one_bit not set to 1 when it should be
WARNING: slice header invalid
Segmentation fault(core dumped)
```

AddressSanitizer output

```
=================================================================
==1960598==ERROR: AddressSanitizer: SEGV on unknown address 0x00009fff8000 (pc 0x7f65de25eac3 bp 0x61b000001c80 sp 0x7ffe41764b90 T0)
==1960598==The signal is caused by a READ memory access.
    #0 0x7f65de25eac2 in slice_segment_header::read(bitreader*, decoder_context*, bool*) /home/dh/sda3/libde265-master/libde265-master/libde265/slice.cc:390
    #1 0x7f65de14837a in decoder_context::read_slice_NAL(bitreader&, NAL_unit*, nal_header&) /home/dh/sda3/libde265-master/libde265-master/libde265/decctx.cc:626
    #2 0x7f65de14a839 in decoder_context::decode_NAL(NAL_unit*) /home/dh/sda3/libde265-master/libde265-master/libde265/decctx.cc:1230
    #3 0x7f65de14be1e in decoder_context::decode(int*) /home/dh/sda3/libde265-master/libde265-master/libde265/decctx.cc:1318
    #4 0x55d4ecf488fd in main /home/dh/sda3/libde265-master/libde265-master/dec265/dec265.cc:764
    #5 0x7f65ddc9a0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)
    #6 0x55d4ecf4b76d in _start (/home/dh/sda3/libde265-master/libde265-master/dec265+0xa76d)

AddressSanitizer can not provide additional info.
SUMMARY: AddressSanitizer: SEGV /home/dh/sda3/libde265-master/libde265-master/libde265/slice.cc:390 in slice_segment_header::read(bitreader*, decoder_context*, bool*)
==1960598==ABORTING
```

gdb info

```
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
WARNING: end_of_sub_stream_one_bit not set to 1 when it should be
WARNING: slice header invalid

Program received signal SIGSEGV, Segmentation fault.
[----------------------------------registers-----------------------------------]
RAX: 0x0
RBX: 0xffffffffffffff90
RCX: 0x617000000090 --> 0x100000000 --> 0x0
RDX: 0xc2e00000013 --> 0x0
RSI: 0x20000000 ('')
RDI: 0x617000000098 --> 0x100000001 --> 0x0
RBP: 0x61b000001c80 --> 0xbebebebe00000000
RSP: 0x7fffffff3570 --> 0x0
RIP: 0x7ffff73abac3 (<slice_segment_header::read(bitreader*, decoder_context*, bool*)+2387>:    movzx  r14d,BYTE PTR [rsi+0x7fff8000])
R8 : 0xfffff8f8 --> 0x0
R9 : 0x7
R10: 0x9 ('\t')
R11: 0xfffffffe6c8 --> 0x0
R12: 0x7ffff31ff800 --> 0xbebebebebebebebe
R13: 0x7fffffff3a40 --> 0x62e000078405 --> 0xbebebebebebebebe
R14: 0xffffe641bdb --> 0x0
R15: 0x555555569bd0 --> 0x7ffff31ff800 --> 0xbebebebebebebebe
EFLAGS: 0x10216 (carry PARITY ADJUST zero sign trap INTERRUPT direction overflow)
[-------------------------------------code-------------------------------------]
   0x7ffff73abab6 <slice_segment_header::read(bitreader*, decoder_context*, bool*)+2374>:       mov    rsi,QWORD PTR [rcx+0x8]
   0x7ffff73ababa <slice_segment_header::read(bitreader*, decoder_context*, bool*)+2378>:       mov    QWORD PTR [rsp+0x10],rsi
   0x7ffff73ababf <slice_segment_header::read(bitreader*, decoder_context*, bool*)+2383>:       shr    rsi,0x3
=> 0x7ffff73abac3 <slice_segment_header::read(bitreader*, decoder_context*, bool*)+2387>:       movzx  r14d,BYTE PTR [rsi+0x7fff8000]
   0x7ffff73abacb <slice_segment_header::read(bitreader*, decoder_context*, bool*)+2395>:       test   r14b,r14b
   0x7ffff73abace <slice_segment_header::read(bitreader*, decoder_context*, bool*)+2398>:
    je     0x7ffff73abad6 <slice_segment_header::read(bitreader*, decoder_context*, bool*)+2406>:           je     0x7ffff73abad6 <slice_segment_header::read(bitreader*,
decoder_context*, bool*)+2406>
```

```
   0x7ffff73abad0 <slice_segment_header::read(bitreader*, decoder_context*, bool*)+2400>:
    jle    0x7ffff73b31dc <slice_segment_header::read(bitreader*, decoder_context*, bool*)+32876>:          jle    0x7ffff73b31dc <slice_segment_header::read(bitreader*,
decoder_context*, bool*)+32876>
   0x7ffff73abad6 <slice_segment_header::read(bitreader*, decoder_context*, bool*)+2406>:       mov    rax,QWORD PTR [rsp+0x10]
[------------------------------------stack-------------------------------------]
0000| 0x7fffffff3570 --> 0x0
0008| 0x7fffffff3578 --> 0x621000000100 --> 0x7ffff7565f30 --> 0x7ffff72719e0 (<decoder_context::~decoder_context()>:   endbr64)
0016| 0x7fffffff3580 --> 0x100000001 --> 0x0
0024| 0x7fffffff3588 --> 0x61b000001c88 --> 0x617000000090 --> 0x100000000 --> 0x0
0032| 0x7fffffff3590 --> 0x7fffffff3780 --> 0x0
0040| 0x7fffffff3598 --> 0x7fffffff3620 --> 0x41b58ab3
0048| 0x7fffffff35a0 --> 0x61b000001ca0 --> 0xbebebe00 --> 0x0
0056| 0x7fffffff35a8 --> 0xfffffffe6c4 --> 0x0
[------------------------------------------------------------------------------]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x00007ffff73abac3 in slice_segment_header::read (
    this=this@entry=0x61b000001c80, br=br@entry=0x7fffffff3a40,
    ctx=ctx@entry=0x621000000100,
    continueDecoding=continueDecoding@entry=0x7fffffff3780)
    at /home/dh/sda3/AFLplusplus/libde265-master/libde265-master-afl++/libde265/slice.cc:390
390       if (!sps->sps_read) {
```

This issue will cause Denial of Service attacks

farindk added a commit that referenced this issue on Apr 5

`fix check for valid PPS idx (#298)` ✕ e83f379

farindk commented on Apr 5 Contributor

Thank you.
Please confirm that the issue is fixed with the above change.

ist199099 commented on Oct 1

This is fixed in the tip of the master branch (commit `b371427`) on Ubuntu 20.04 (with GCC 9.4.0 and Clang 10.0.0) on the x86_64 and aarch64 architectures.

farindk commented on Oct 1 Contributor

@ist199099 Thank you for cross checking this.

farindk closed this as completed on Oct 1

coldtobi commented 4 days ago

According to Debian this is CVE-2021-35452

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

4 participants