New issue

Jump to bottom

A SEGV has occurred when running program dec265 #302

Open **dhbbb** opened this issue on Jun 25, 2021 · 2 comments

dhbbb commented on Jun 25, 2021

Hello,
A SEGV of deblock.cc in function derive_boundaryStrength has occurred when running program dec265 ,

source code

```
283 if ((edgeFlags & transformEdgeMask) &&
284             (img->get_nonzero_coefficient(xDi   ,yDi) ||
285              img->get_nonzero_coefficient(xDiOpp,yDiOpp))) {
286           bS = 1;
287         }
288         else {
289
290           bS = 0;
291
292           const PBMotion& mviP = img->get_mv_info(xDiOpp,yDiOpp);
293           const PBMotion& mviQ = img->get_mv_info(xDi   ,yDi);
294
295           slice_segment_header* shdrP = img->get_SliceHeader(xDiOpp,yDiOpp);
296           slice_segment_header* shdrQ = img->get_SliceHeader(xDi   ,yDi);
297
298           int refPicP0 = mviP.predFlag[0] ? shdrP->RefPicList[0][ mviP.refIdx[0] ] : -1;
299           int refPicP1 = mviP.predFlag[1] ? shdrP->RefPicList[1][ mviP.refIdx[1] ] : -1;
300           int refPicQ0 = mviQ.predFlag[0] ? shdrQ->RefPicList[0][ mviQ.refIdx[0] ] : -1;
301           int refPicQ1 = mviQ.predFlag[1] ? shdrQ->RefPicList[1][ mviQ.refIdx[1] ] : -1;
302
303           bool samePics = ((refPicP0==refPicQ0 && refPicP1==refPicQ1) ||
304                            (refPicP0==refPicQ1 && refPicP1==refPicQ0));
```

Due to incorrect access control, a SEGV caused by a READ memory access occurred at line 298 of the code. This issue can cause a Denial of Service attack.

System info：
Ubuntu 20.04.1 : clang 10.0.0 , gcc 9.3.0

Dec265 v1.0.8

poc.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: CTB outside of image area (concealing stream error...)
WARNING: CTB outside of image area (concealing stream error...)
Segmentation fault(core dumped)
```

AddressSanitizer output

```
AddressSanitizer:DEADLYSIGNAL
=================================================================
==3532158==ERROR: AddressSanitizer: SEGV on unknown address 0x0000000003d0 (pc 0x7f19b4f52978 bp 0x616000001580 sp 0x7fff00e87c20 T0)
==3532158==The signal is caused by a READ memory access.
==3532158==Hint: address points to the zero page.
    #0 0x7f19b4f52977 in derive_boundaryStrength(de265_image*, bool, int, int, int, int) /home/dh/sda3/libde265-master/libde265-master/libde265/deblock.cc:298
    #1 0x7f19b4f56835 in apply_deblocking_filter(de265_image*) /home/dh/sda3/libde265-master/libde265-master/libde265/deblock.cc:1046
    #2 0x7f19b4f7e626 in decoder_context::run_postprocessing_filters_sequential(de265_image*) /home/dh/sda3/libde265-master/libde265-master/libde265/decctx.cc:1880
    #3 0x7f19b4f9baa0 in decoder_context::decode_some(bool*) /home/dh/sda3/libde265-master/libde265-master/libde265/decctx.cc:769
    #4 0x7f19b4f9f95e in decoder_context::decode(int*) /home/dh/sda3/libde265-master/libde265-master/libde265/decctx.cc:1329
    #5 0x55704ed8c8fd in main /home/dh/sda3/libde265-master/libde265-master/dec265/dec265.cc:764
    #6 0x7f19b4aee0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)
    #7 0x55704ed8f76d 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/deblock.cc:298 in derive_boundaryStrength(de265_image*, bool, int, int, int, int)
==3532158==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: non-existing reference picture accessed
WARNING: non-existing reference picture accessed
WARNING: non-existing reference picture accessed
WARNING: non-existing reference picture accessed
WARNING: non-existing reference picture accessed
WARNING: non-existing reference picture accessed
WARNING: CTB outside of image area (concealing stream error...)
WARNING: CTB outside of image area (concealing stream error...)

Program received signal SIGSEGV, Segmentation fault.
[----------------------------------registers-----------------------------------]
RAX: 0x0
RBX: 0x2
RCX: 0x61b000001580 --> 0xbebebebe00000000
RDX: 0x0
RSI: 0x7a ('z')
RDI: 0x3d0
RBP: 0x616000001580 --> 0xbebebebe00000007
RSP: 0x7fffffff36e0 --> 0x3000000000 --> 0x0
RIP: 0x7ffff724b978 (<derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6024>:    mov    ebx,DWORD PTR [r9+r15*4+0x3b8])
R8 : 0x3
R9 : 0x0
R10: 0x6330000d6800 --> 0x8ffff00000101
R11: 0x6330000d6200 --> 0x60101
R12: 0x0
R13: 0xfffffffffffffff90
R14: 0x7ffff31ff800 --> 0xbebebebebebebebe
R15: 0x6
EFLAGS: 0x10246 (carry PARITY adjust ZERO sign trap INTERRUPT direction overflow)
[-------------------------------------code-------------------------------------]
   0x7ffff724b96e <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6014>:
    jl     0x7ffff724b978 <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6024>
   0x7ffff724b970 <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6016>:       test   dl,dl
   0x7ffff724b972 <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6018>:
    jne    0x7ffff724dd87 <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+15255>
=> 0x7ffff724b978 <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6024>:       mov    ebx,DWORD PTR [r9+r15*4+0x3b8]
   0x7ffff724b980 <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6032>:       mov    edx,0x376d
   0x7ffff724b985 <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6037>:       mov    eax,0xafce
   0x7ffff724b98a <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6042>:       lea    r15,[r11+0x1]
   0x7ffff724b98e <derive_boundaryStrength(de265_image*, bool, int, int, int, int)+6046>:       mov    rdi,r15
[------------------------------------stack-------------------------------------]
0000| 0x7fffffff36e0 --> 0x3000000000 --> 0x0
0008| 0x7fffffff36e8 --> 0x6160000016f8 --> 0x4000000080 --> 0x0
0016| 0x7fffffff36f0 --> 0x6160000016e8 --> 0x625000057900 --> 0x0
0024| 0x7fffffff36f8 --> 0xa000000080 --> 0x0
0032| 0x7fffffff3700 --> 0x1
0040| 0x7fffffff3708 --> 0xbf000000c0 --> 0x0
0048| 0x7fffffff3710 --> 0x61600000167c --> 0x4000000003 --> 0x0
0056| 0x7fffffff3718 --> 0xff00f800 --> 0x0
[------------------------------------------------------------------------------]
Legend: code, data, rodata, value
Stopped reason: SIGSEGV
0x00007ffff724b978 in derive_boundaryStrength (img=img@entry=0x616000001580,
    vertical=vertical@entry=0x0, yStart=yStart@entry=0x0,
    yEnd=<optimized out>, xStart=xStart@entry=0x0, xEnd=<optimized out>)
    at /home/dh/sda3/AFLplusplus/libde265-master/libde265-master-afl++/libde265/deblock.cc:298
298                int refPicP0 = mviP.predFlag[0] ? shdrP->RefPicList[0][ mviP.refIdx[0] ] : -1;
```

stevebeattie commented on Jan 12

This issue was assigned CVE-2021-36411.

farindk added a commit that referenced this issue on Apr 5

`fix reading invalid images where shdr references are NULL in part of …` ··· `45904e5`

farindk commented on Apr 5 — Contributor

Thanks.
Please confirm that issue is fixed with above change.

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

3 participants