

[New issue](#)[Jump to bottom](#)

# Unknown crash in sse-motion.cc: ff\_hevc\_put\_hevc\_qpel\_h\_3\_v\_3\_sse #336

[Open](#) FDU-Sec opened this issue on Oct 10 · 0 comments

FDU-Sec commented on Oct 10

## Description

SUMMARY: AddressSanitizer: unknown-crash (/libde265/build/libde265/liblibde265.so+0x28fa79) in ff\_hevc\_put\_hevc\_qpel\_h\_3\_v\_3\_sse(short\*, long, unsigned char const\*, long, int, int, short\*)

## Version

```
$ ./dec265 -h
dec265 v1.0.8
-----
usage: dec265 [options] videofile.bin
The video file must be a raw bitstream, or a stream with NAL units (option -n).

options:
  -q, --quiet           do not show decoded image
  -t, --threads N       set number of worker threads (0 - no threading)
  -c, --check-hash      perform hash check
  -n, --nal             input is a stream with 4-byte length prefixed NAL units
  -f, --frames N        set number of frames to process
  -o, --output          write YUV reconstruction
  -d, --dump            dump headers
  -0, --noaccel         do not use any accelerated code (SSE)
  -v, --verbose         increase verbosity level (up to 3 times)
  -L, --no-logging      disable logging
  -B, --write-bytestream FILENAME write raw bytestream (from NAL input)
  -m, --measure YUV     compute PSNRs relative to reference YUV
  -T, --highest-TID     select highest temporal sublayer to decode
                        --disable-deblocking disable deblocking filter
                        --disable-sao       disable sample-adaptive offset filter
  -h, --help           show help
```

## Replay

```
git clone https://github.com/strukturag/libde265.git
cd libde265
mkdir build
cd build
cmake ../ -DCMAKE_CXX_FLAGS="-fsanitize=address"
make -j$(nproc)
./dec265/dec265 poc2
```

## ASAN

```
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: non-existing PPS referenced
WARNING: non-existing PPS referenced
WARNING: pps header invalid
WARNING: non-existing PPS referenced
WARNING: pps header invalid
WARNING: slice header invalid
WARNING: pps header invalid
=====
==35433==ERROR: AddressSanitizer: unknown-crash on address 0x7f812fbf9806 at pc 0x7f812e8b6a7a bp 0x7
READ of size 16 at 0x7f812fbf9806 thread T0
#0 0x7f812e8b6a79 in ff_hevc_put_hevc_qpel_h_3_v_3_sse(short*, long, unsigned char const*, long,
#1 0x7f812e79e37d in acceleration_functions::put_hevc_qpel(short*, long, void const*, long, int,
#2 0x7f812e79f8ab in void mc_luma<unsigned char>(base_context const*, seq_parameter_set const*, i
#3 0x7f812e790995 in generate_inter_prediction_samples(base_context*, slice_segment_header const*
#4 0x7f812e79d90f in decode_prediction_unit(base_context*, slice_segment_header const*, de265_ima
#5 0x7f812e7d87e3 in read_prediction_unit(thread_context*, int, int, int, int, int, int, int, int
#6 0x7f812e7da3fe in read_coding_unit(thread_context*, int, int, int, int) (/libde265/build/libde
#7 0x7f812e7db250 in read_coding_quadtree(thread_context*, int, int, int, int) (/libde265/build/l
#8 0x7f812e7db163 in read_coding_quadtree(thread_context*, int, int, int, int) (/libde265/build/l
#9 0x7f812e7db163 in read_coding_quadtree(thread_context*, int, int, int, int) (/libde265/build/l
#10 0x7f812e7d2726 in read_coding_tree_unit(thread_context*) (/libde265/build/libde265/liblibde26
#11 0x7f812e7db9ea in decode_substream(thread_context*, bool, bool) (/libde265/build/libde265/lib
#12 0x7f812e7dd70f in read_slice_segment_data(thread_context*) (/libde265/build/libde265/liblibde
#13 0x7f812e73c6d2 in decoder_context::decode_slice_unit_sequential(image_unit*, slice_unit*) (/l
#14 0x7f812e73cec1 in decoder_context::decode_slice_unit_parallel(image_unit*, slice_unit*) (/lib
#15 0x7f812e73bc0f in decoder_context::decode_some(bool*) (/libde265/build/libde265/liblibde265.s
#16 0x7f812e73b93d in decoder_context::read_slice_NAL(bitreader&, NAL_unit*, nal_header&) (/libde
#17 0x7f812e73e43e in decoder_context::decode_NAL(NAL_unit*) (/libde265/build/libde265/liblibde26
#18 0x7f812e73eab3 in decoder_context::decode(int*) (/libde265/build/libde265/liblibde265.so+0x11
#19 0x7f812e725e95 in de265_decode (/libde265/build/libde265/liblibde265.so+0xf9e95)
#20 0x55b9f5596bc9 in main (/libde265/build/dec265/dec265+0x6bc9)
#21 0x7f812e257c86 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21c86)
#22 0x55b9f55949b9 in _start (/libde265/build/dec265/dec265+0x49b9)

0x7f812fbf9810 is located 0 bytes to the right of 131088-byte region [0x7f812fbd9800,0x7f812fbf9810)
allocated by thread T0 here:
#0 0x7f812ec4e790 in posix_memalign (/usr/lib/x86_64-linux-gnu/libasan.so.4+0xdf790)
#1 0x7f812e7771cb in ALLOC_ALIGNED(unsigned long, unsigned long) (/libde265/build/libde265/liblib
#2 0x7f812e77792a in de265_image_get_buffer(void*, de265_image_spec*, de265_image*, void*) (/lib
#3 0x7f812e779d1a in de265_image::alloc_image(int, int, de265_chroma, std::shared_ptr<seq_paramet
#4 0x7f812e75e0cc in decoded_picture_buffer::new_image(std::shared_ptr<seq_parameter_set const>,
```

```
#5 0x7f812e73f824 in decoder_context::generate_unavailable_reference_picture(seq_parameter_set co
#6 0x7f812e742332 in decoder_context::process_reference_picture_set(slice_segment_header*) (/libd
#7 0x7f812e745d70 in decoder_context::process_slice_segment_header(slice_segment_header*, de265_e
#8 0x7f812e73b246 in decoder_context::read_slice_NAL(bitreader&, NAL_unit*, nal_header&) (/libde2
#9 0x7f812e73e43e in decoder_context::decode_NAL(NAL_unit*) (/libde265/build/libde265/liblibde265
#10 0x7f812e73eab3 in decoder_context::decode(int*) (/libde265/build/libde265/liblibde265.so+0x11
#11 0x7f812e725e95 in de265_decode (/libde265/build/libde265/liblibde265.so+0xf9e95)
#12 0x55b9f5596bc9 in main (/libde265/build/dec265/dec265+0x6bc9)
#13 0x7f812e257c86 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21c86)
```

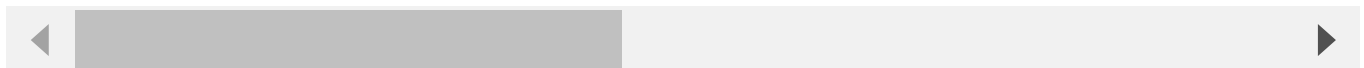
SUMMARY: AddressSanitizer: unknown-crash (/libde265/build/libde265/liblibde265.so+0x28fa79) in ff\_hev

Shadow bytes around the buggy address:

```
0x0ff0a5f772b0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0ff0a5f772c0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0ff0a5f772d0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0ff0a5f772e0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0ff0a5f772f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0ff0a5f77300:[00]00 fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0ff0a5f77310: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0ff0a5f77320: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0ff0a5f77330: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0ff0a5f77340: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0ff0a5f77350: fa fa 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
Freed heap region:      fd
Stack left redzone:     f1
Stack mid redzone:      f2
Stack right redzone:    f3
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
Left alloca redzone:    ca
Right alloca redzone:   cb
==35433==ABORTING
```



## POC

<https://github.com/FDU-Sec/poc/blob/main/libde265/poc2>

## Environment

Ubuntu 16.04  
Clang 10.0.1  
gcc 5.5

## Credit

Peng Deng ([Fudan University](#))

### Assignees

No one assigned

---

### Labels

None yet

---

### Projects

None yet

---

### Milestone

No milestone

---

### Development

No branches or pull requests

---

1 participant

