

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

Heap-buffer-overflow in fallback-motion.cc: in void put_qpel_fallback<unsigned short> #342

Open FDU-Sec opened this issue on Oct 10 · 0 comments

FDU-Sec commented on Oct 10

Description

Heap-buffer-overflow (/libde265/build/libde265/liblibde265.so+0x14b860) in void put_qpel_fallback(short*, long, unsigned short const*, long, int, int, short*, int, int, int)

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 poc8-1
./dec265/dec265 poc8-2
./dec265/dec265 poc8-3
./dec265/dec265 poc8-4
```

ASAN

WARNING: end_of_sub_stream_one_bit not **set** to 1 when it should be

WARNING: maximum number of reference pictures exceeded

WARNING: CTB outside of image area (concealing stream error...)

=====

==55253==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x62f00001ac80 at pc 0x7f7d9b220861
READ of size 2 at 0x62f00001ac80 thread T0

```
#0 0x7f7d9b220860 in void put_qpel_fallback<unsigned short>(short*, long, unsigned short const*,
#1 0x7f7d9b21c05c in put_qpel_0_3_fallback_16(short*, long, unsigned short const*, long, int, int
#2 0x7f7d9b24c40d in acceleration_functions::put_hevc_qpel(short*, long, void const*, long, int,
#3 0x7f7d9b24cee6 in void mc_luma<unsigned short>(base_context const*, seq_parameter_set const*,
#4 0x7f7d9b23e837 in generate_inter_prediction_samples(base_context*, slice_segment_header const*
#5 0x7f7d9b24b90f in decode_prediction_unit(base_context*, slice_segment_header const*, de265_ima
#6 0x7f7d9b2867e3 in read_prediction_unit(thread_context*, int, int, int, int, int, int, int, int
#7 0x7f7d9b288333 in read_coding_unit(thread_context*, int, int, int, int) (/libde265/build/libde
#8 0x7f7d9b289250 in read_coding_quadtree(thread_context*, int, int, int, int) (/libde265/build/l
#9 0x7f7d9b289091 in read_coding_quadtree(thread_context*, int, int, int, int) (/libde265/build/l
#10 0x7f7d9b280726 in read_coding_tree_unit(thread_context*) (/libde265/build/libde265/liblibde26
#11 0x7f7d9b2899ea in decode_substream(thread_context*, bool, bool) (/libde265/build/libde265/lib
#12 0x7f7d9b28b70f in read_slice_segment_data(thread_context*) (/libde265/build/libde265/liblibde
#13 0x7f7d9b1ea6d2 in decoder_context::decode_slice_unit_sequential(image_unit*, slice_unit*) (/l
#14 0x7f7d9b1eaec1 in decoder_context::decode_slice_unit_parallel(image_unit*, slice_unit*) (/lib
#15 0x7f7d9b1e9c0f in decoder_context::decode_some(bool*) (/libde265/build/libde265/liblibde265.s
#16 0x7f7d9b1e993d in decoder_context::read_slice_NAL(bitreader&, NAL_unit*, nal_header&) (/libde
#17 0x7f7d9b1ec43e in decoder_context::decode_NAL(NAL_unit*) (/libde265/build/libde265/liblibde26
#18 0x7f7d9b1ecab3 in decoder_context::decode(int*) (/libde265/build/libde265/liblibde265.so+0x11
#19 0x7f7d9b1d3e95 in de265_decode (/libde265/build/libde265/liblibde265.so+0xfef95)
#20 0x55ae31f1cbc9 in main (/libde265/build/dec265/dec265+0x6bc9)
#21 0x7f7d9ad05c86 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21c86)
#22 0x55ae31f1a9b9 in _start (/libde265/build/dec265/dec265+0x49b9)
```

0x62f00001ac80 is located 112 bytes to the right of 51216-byte region [0x62f00000e400,0x62f00001ac10)
allocated by thread T0 here:

```
#0 0x7f7d9b6fc790 in posix_memalign (/usr/lib/x86_64-linux-gnu/libasan.so.4+0xdf790)
#1 0x7f7d9b2251cb in ALLOC_ALIGNED(unsigned long, unsigned long) (/libde265/build/libde265/liblib
#2 0x7f7d9b22592a in de265_image_get_buffer(void*, de265_image_spec*, de265_image*, void*) (/libd
#3 0x7f7d9b227d1a in de265_image::alloc_image(int, int, de265_chroma, std::shared_ptr<seq_paramet
#4 0x7f7d9b20c0cc in decoded_picture_buffer::new_image(std::shared_ptr<seq_parameter_set const>,
#5 0x7f7d9b1ed824 in decoder_context::generate_unavailable_reference_picture(seq_parameter_set co
#6 0x7f7d9b1f0332 in decoder_context::process_reference_picture_set(slice_segment_header*) (/libd
#7 0x7f7d9b1f3d70 in decoder_context::process_slice_segment_header(slice_segment_header*, de265_e
```

```
#8 0x7f7d9b1e9246 in decoder_context::read_slice_NAL(bitreader&, NAL_unit*, nal_header&) (/libde2
#9 0x7f7d9b1ec43e in decoder_context::decode_NAL(NAL_unit*) (/libde265/build/libde265/liblibde265
#10 0x7f7d9b1ecab3 in decoder_context::decode(int*) (/libde265/build/libde265/liblibde265.so+0x11
#11 0x7f7d9b1d3e95 in de265_decode (/libde265/build/libde265/liblibde265.so+0xf9e95)
#12 0x55ae31f1cbc9 in main (/libde265/build/dec265/dec265+0x6bc9)
#13 0x7f7d9ad05c86 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21c86)
```

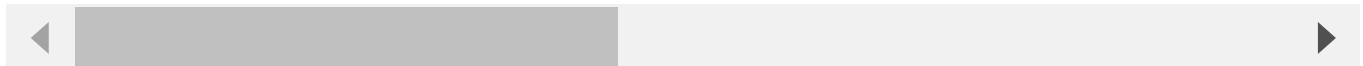
SUMMARY: AddressSanitizer: heap-buffer-overflow (/libde265/build/libde265/liblibde265.so+0x14b860) in
Shadow bytes around the buggy address:

```
0x0c5e7fffb540: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c5e7fffb550: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c5e7fffb560: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c5e7fffb570: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c5e7fffb580: 00 00 fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
=>0x0c5e7fffb590:[fa]fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c5e7fffb5a0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c5e7fffb5b0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c5e7fffb5c0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c5e7fffb5d0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c5e7fffb5e0: 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
```

==55253==ABORTING



POC

<https://github.com/FDU-Sec/poc/blob/main/libde265/poc8-1>

<https://github.com/FDU-Sec/poc/blob/main/libde265/poc8-2>

<https://github.com/FDU-Sec/poc/blob/main/libde265/poc8-3>

<https://github.com/FDU-Sec/poc/blob/main/libde265/poc8-4>

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

