#8859 closed defect (duplicate)

Opened 2 years ago
Closed 2 years ago

A heap-buffer-overflow in aacdec_template.c:543:15

Reported by:	Zhou Anshunkang	Owned by:	
Priority:	normal	Component:	avcodec
Version:	git-master	Keywords:	aac
Cc:		Blocked By:	
Blocking:		Reproduced by developer:	no
Analyzed by developer:	no		

Description

System info

Ubuntu x86_64, clang 6.0, ffmpeg (git-master https://github.com/FFmpeg/FFmpeg/commit/3fc3d712a99cf39f69a2258b48cbc81fa8ae5471)

Configure

```
./configure --disable-shared --enable-debug=3 --disable-ffplay --disable-ffprobe -
```

Command line

```
./ffmpeg -y -f mov /dev/null -i @@
```

AddressSanitizer

```
=================================================================
==35580==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x624000008308 a
READ of size 8 at 0x624000008308 thread T0
    #0 0x30dcdfb  (/home/seviezhou/ffmpeg/ffmpeg+0x30dcdfb)
    #1 0x30d1314  (/home/seviezhou/ffmpeg/ffmpeg+0x30d1314)
    #2 0x30e7f96  (/home/seviezhou/ffmpeg/ffmpeg+0x30e7f96)
    #3 0x30c7dbc  (/home/seviezhou/ffmpeg/ffmpeg+0x30c7dbc)
    #4 0x179945c  (/home/seviezhou/ffmpeg/ffmpeg+0x179945c)
    #5 0x1798a55  (/home/seviezhou/ffmpeg/ffmpeg+0x1798a55)
    #6 0x145def6  (/home/seviezhou/ffmpeg/ffmpeg+0x145def6)
    #7 0x1451f5d  (/home/seviezhou/ffmpeg/ffmpeg+0x1451f5d)
    #8 0x519abb  (/home/seviezhou/ffmpeg/ffmpeg+0x519abb)
    #9 0x5179a6  (/home/seviezhou/ffmpeg/ffmpeg+0x5179a6)
    #10 0x516e1b  (/home/seviezhou/ffmpeg/ffmpeg+0x516e1b)
    #11 0x5839c2  (/home/seviezhou/ffmpeg/ffmpeg+0x5839c2)
    #12 0x7f6ec8c51b96 in __libc_start_main /build/glibc-OTsEL5/glibc-2.27/csu/../
    #13 0x41e179 in _init (/home/seviezhou/ffmpeg/ffmpeg+0x41e179)

Address 0x624000008308 is a wild pointer.
SUMMARY: AddressSanitizer: heap-buffer-overflow (/home/seviezhou/ffmpeg/ffmpeg+0x3
Shadow bytes around the buggy address:
  0x0c487fff9010: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff9020: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff9030: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff9040: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff9050: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
=>0x0c487fff9060: fa[fa]fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff9070: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff9080: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff9090: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff90a0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff90b0: 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
==35580==ABORTING
```

Attachments (1)

- heap-overflow-ffmpeg-JIT(227 bytes) - added by Carl Eugen Hoyos 2 years ago.

Change History (7)

comment:1 by Carl Eugen Hoyos, 2 years ago — in reply to: description

Component: ffmpeg → avcodec
Keywords: aac added

Replying to ~~seviezhou~~:

> ```
> ./configure --disable-shared --enable-debug=3 --disable-ffplay --disable-ffprobe -
> ```

This will get more readable if you use `./configure --toolchain=clang-asan && make ffmpeg_g`, feel free to add `--disable-asm`.

> ```
> ./ffmpeg -y -f mov /dev/null -i @@
> ```

This is unfortunately useless, use `ffmpeg_g` instead and post everything that gets printed, not only the part you consider important (we disagree).

Please do not use zip here and please explain what "JIT" means.

by Carl Eugen Hoyos, 2 years ago

Attachment: *heap-overflow-ffmpeg-JIT*added

comment:2 by Cigaes, 2 years ago

Just a small question: what is this "JIT code" you are referring to?

comment:3 by Zhou Anshunkang, 2 years ago

Just because although I enabled debug information and disable inline assembly, I still cannot get any source line information from the backtrace. So I think the crash point might not in the source code, but in the assembly code. So I guess that the crash point is in some "just in time generated" code. Any ideas about this?

comment:4 by Zhou Anshunkang, 2 years ago

Here is the output of ffmpeg_g, I am sorry for my previous post:

```
ffmpeg version N-98801-g3fc3d712a9 Copyright (c) 2000-2020 the FFmpeg developers
  built with clang version 6.0.0-1ubuntu2 (tags/RELEASE_600/final)
  configuration: --disable-shared --enable-debug=3 --disable-ffplay --disable-ffprob
  libavutil      56. 58.100 / 56. 58.100
  libavcodec     58.101.100 / 58.101.100
  libavformat    58. 51.100 / 58. 51.100
  libavdevice    58. 11.101 / 58. 11.101
  libavfilter     7. 87.100 /  7. 87.100
  libswscale      5.  8.100 /  5.  8.100
  libswresample   3.  8.100 /  3.  8.100
[aac @ 0x61b000000080] Format aac detected only with low score of 1, misdetection po
[aac @ 0x619000000580] More than one AAC RDB per ADTS frame is not implemented. Upda
[aac @ 0x61b000000080] Packet corrupt (stream = 0, dts = NOPTS).
[aac @ 0x619000000580] Error decoding AAC frame header.
[aac @ 0x619000000580] Sample rate index in program config element does not match th
=================================================================
==36919==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x6240000055d0 at
READ of size 8 at 0x6240000055d0 thread T0
    #0 0x27e9427 in che_configure /home/seviezhou/ffmpeg/libavcodec/aacdec_template
    #1 0x27db93d in output_configure /home/seviezhou/ffmpeg/libavcodec/aacdec_templa
    #2 0x27ef51c in aac_decode_frame_int /home/seviezhou/ffmpeg/libavcodec/aacdec_te
    #3 0x27d7843 in aac_decode_frame /home/seviezhou/ffmpeg/libavcodec/aacdec_templa
    #4 0x12659b8 in decode_simple_internal /home/seviezhou/ffmpeg/libavcodec/decode
    #5 0x12659b8 in decode_simple_receive_frame /home/seviezhou/ffmpeg/libavcodec/de
    #6 0x12659b8 in decode_receive_frame_internal /home/seviezhou/ffmpeg/libavcodec/
    #7 0x12652cd in avcodec_send_packet /home/seviezhou/ffmpeg/libavcodec/decode.c:
    #8 0xfabadf in try_decode_frame /home/seviezhou/ffmpeg/libavformat/utils.c:3111
    #9 0xfa3054 in avformat_find_stream_info /home/seviezhou/ffmpeg/libavformat/util
    #10 0x5181fa in open_input_file /home/seviezhou/ffmpeg/fftools/ffmpeg_opt.c:118
    #11 0x516d6a in open_files /home/seviezhou/ffmpeg/fftools/ffmpeg_opt.c:3303:15
    #12 0x516795 in ffmpeg_parse_options /home/seviezhou/ffmpeg/fftools/ffmpeg_opt.
    #13 0x555d8f in main /home/seviezhou/ffmpeg/fftools/ffmpeg.c:4850:11
    #14 0x7fd409933b96 in __libc_start_main /build/glibc-OTsEL5/glibc-2.27/csu/../c
    #15 0x41df49 in _start (/home/seviezhou/ffmpeg/ffmpeg_g+0x41df49)

Address 0x6240000055d0 is a wild pointer.
SUMMARY: AddressSanitizer: heap-buffer-overflow /home/seviezhou/ffmpeg/libavcodec/aa
Shadow bytes around the buggy address:
  0x0c487fff8a60: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff8a70: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff8a80: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff8a90: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff8aa0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
=>0x0c487fff8ab0: fa fa fa fa fa fa fa fa fa fa[fa]fa fa fa fa fa
  0x0c487fff8ac0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff8ad0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff8ae0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff8af0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c487fff8b00: 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
==36919==ABORTING
```

comment:5 by Zhou Anshunkang, 2 years ago

Summary: A heap-buffer-overflow in FFmpeg JIT code → A heap-buffer-overflow in aacdec_template.c:543:15

comment:6 by Carl Eugen Hoyos, 2 years ago

Resolution: → duplicate
Status: new → closed

Fixed by Jan Ekström in d6f293353c94c7ce200f6e0975ae3de49787f91f

Note: See TracTickets for help on using tickets.