#7996 closed defect (duplicate)

Opened 3 years ago
Closed 19 months ago
Last modified 19 months ago

Division by zero at libavcodec/lpc.h:155

Reported by:	Suhwan	Owned by:	
Priority:	normal	Component:	undetermined
Version:	git-master	Keywords:	ubsan asan
Cc:	Michael Niedermayer	Blocked By:	
Blocking:		Reproduced by developer:	no
Analyzed by developer:	no		

Description

Summary of the bug:
There's division by zero at libavcodec/lpc.h:155 and heap buffer overflow at libavcodec/zmbvenc.c:97:30.

How to reproduce:

```
% ffmpeg_g -y -r 48 -i tmp.wmv -map 0 -c:s:4 zmbv -c:v zmbv -disposition:a:19 fits

ffmpeg version : N-94163-g664a27ea40
built with clang version 9.0.0
```

```
135 static inline void compute_ref_coefs(const LPC_TYPE *autoc, int max_order,
136                                        LPC_TYPE *ref, LPC_TYPE *error)
137 {
138     int i, j;
139     LPC_TYPE err;
140     LPC_TYPE gen0[MAX_LPC_ORDER], gen1[MAX_LPC_ORDER];
141
142     for (i = 0; i < max_order; i++)
143         gen0[i] = gen1[i] = autoc[i + 1];
144
145     err    = autoc[0];
146     ref[0] = -gen1[0] / err;
147     err   +=  gen1[0] * ref[0];
148     if (error)
149         error[0] = err;
150     for (i = 1; i < max_order; i++) {
151         for (j = 0; j < max_order - i; j++) {
152             gen1[j] = gen1[j + 1] + ref[i - 1] * gen0[j];
153             gen0[j] = gen1[j + 1] * ref[i - 1] + gen0[j];
154         }
155         ref[i] = -gen1[0] / err;
156         err   +=  gen1[0] * ref[i];
157         if (error)
158             error[i] = err;
159     }
160 }
```

Attachments (2)

- gdb_log_7996(8.1 KB) - added by Suhwan 3 years ago.
- tmp.wmv(444.8 KB) - added by Suhwan 3 years ago.

Change History (7)

by Suhwan, 3 years ago

Attachment: *gdb_log_7996*added

by Suhwan, 3 years ago

Attachment: *tmp.wmv*added

comment:1 by Suhwan, 3 years ago

```
ffmpeg version N-94906-gcb8d6a4e3e Copyright (c) 2000-2019 the FFmpeg developers
  built with clang version 6.0.0-1ubuntu2 (tags/RELEASE_600/final)
  configuration: --cc=clang --cxx=clang++ --ld=clang --enable-debug --toolchain=cla
  libavutil      56. 35.100 / 56. 35.100
  libavcodec     58. 56.101 / 58. 56.101
  libavformat    58. 32.104 / 58. 32.104
  libavdevice    58.  9.100 / 58.  9.100
  libavfilter     7. 58.102 /  7. 58.102
  libswscale      5.  6.100 /  5.  6.100
  libswresample   3.  6.100 /  3.  6.100
Guessed Channel Layout for Input Stream #0.1 : mono
Input #0, asf, from 'tmp.wmv':
  Metadata:
    encoder         : Lavf57.66.105
  Duration: 00:00:05.63, start: 0.000000, bitrate: 647 kb/s
    Stream #0:0: Video: wmv2 (WMV2 / 0x32564D57), yuv420p, 560x320, SAR 1:1 DAR 7:4,
    Stream #0:1(eng): Audio: wmav2 (a[1][0][0] / 0x0161), 48000 Hz, mono, fltp, 128
Stream mapping:
  Stream #0:0 -> #0:0 (wmv2 (native) -> zmbv (native))
  Stream #0:1 -> #0:1 (wmav2 (native) -> aac (native))
Press [q] to stop, [?] for help
[aac @ 0xab799c0] Using a PCE to encode channel layout "2.1"
[aac @ 0xab799c0] Too many bits 19669.333333 > 18432 per frame requested, clamping t
libavcodec/lpc.h:155:27: runtime error: division by zero
[mov @ 0xab99780] Using MS style video codec tag, the file may be unplayable!
Output #0, mov, to 'tmp_.mov':
  Metadata:
    encoder         : Lavf58.32.104
    Stream #0:0: Video: zmbv, bgr0, 560x320 [SAR 1:1 DAR 7:4], q=2-31, 501 kb/s, 48
    Metadata:
      encoder         : Lavc58.56.101 zmbv
    Stream #0:1(eng): Audio: aac (LC) (mp4a / 0x6134706D), 48000 Hz, 2.1, fltp, 864
    Metadata:
      encoder         : Lavc58.56.101 aac
frame=   34 fps=5.3 q=-0.0 Lsize=    1221kB time=00:00:01.17 bitrate=8547.2kbits/s s
video:1181kB audio:38kB subtitle:0kB other streams:0kB global headers:0kB muxing ov
[aac @ 0xab799c0] Qavg: 64919.828
```

comment:2 by Michael Niedermayer, 19 months ago

Cc: Michael Niedermayer added
Resolution: → fixed
Status: new → closed

The division by 0 happens in floating point and divisions by 0 in floating point are generally not bugs as such. If you disagree and see an issue with this division then please reopen this ticket

Completely unrelated to this the sample triggers a out of array read that has been fixed by def04022f4a7058f99e669bfd978d431d79aec18 so iam marking this as fixed

comment:3 by Michael Niedermayer, 19 months ago

Actually the only bug in this is a duplicate of ~~#7980~~

comment:4 by Michael Niedermayer, 19 months ago

Resolution: fixed → duplicate

comment:5 by Michael Niedermayer, 19 months ago

I will post a patch to ffmpeg-devel to avoid the floating point division.

Note: See TracTickets for help on using tickets.