#8313 closed defect (fixed)

memory leaks in filter_frame()

Reported by:	Suhwan	Owned by:	
Priority:	important	Component:	undetermined
Version:	git-master	Keywords:	
Cc:		Blocked By:	
Blocking:		Reproduced by developer:	no
Analyzed by developer:	no		

Description

Summary of the bug:
There are memory leaks in filter_frame()
How to reproduce:

```
% ffmpeg_g -y -i $PoC -filter_complex tile -target dvd -loglevel 0 -vbsf mpeg4_unp

ffmpeg version N-95464-g7056ddc0e0 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
```

◀   ▶

Here's Valgrind log

```
==47004== HEAP SUMMARY:
==47004==     in use at exit: 323,351 bytes in 7 blocks
==47004==   total heap usage: 2,604 allocs, 2,597 frees, 2,239,057 bytes allocated
==47004==
==47004== 323,319 (536 direct, 322,783 indirect) bytes in 1 blocks are definitely
==47004==    at 0x9FE3E76: memalign (in /usr/lib/valgrind/vgpreload_memcheck-amd64
==47004==    by 0x9FE3F91: posix_memalign (in /usr/lib/valgrind/vgpreload_memcheck
==47004==    by 0x592F079: av_malloc (mem.c:87)
==47004==    by 0x592F079: av_mallocz (mem.c:238)
==47004==    by 0x590320D: av_frame_alloc (frame.c:191)
==47004==    by 0x64FD65: ff_frame_pool_get (framepool.c:201)
==47004==    by 0xFB8D7C: ff_default_get_video_buffer (video.c:90)
==47004==    by 0xD892DB: filter_frame (vf_tile.c:203)
==47004==    by 0x5D002C: ff_filter_frame_framed (avfilter.c:1084)
==47004==    by 0x5D002C: ff_filter_frame_to_filter (avfilter.c:1232)
==47004==    by 0x5D002C: ff_filter_activate_default (avfilter.c:1281)
==47004==    by 0x5D002C: ff_filter_activate (avfilter.c:1443)
==47004==    by 0x5F4434: push_frame (buffersrc.c:187)
==47004==    by 0x5F4434: av_buffersrc_add_frame_internal (buffersrc.c:261)
==47004==    by 0x5F300D: av_buffersrc_add_frame_flags (buffersrc.c:170)
==47004==    by 0x4CAD7F: ifilter_send_frame (ffmpeg.c:2186)
==47004==    by 0x4CAD7F: send_frame_to_filters (ffmpeg.c:2260)
==47004==    by 0x4A07DB: decode_video (ffmpeg.c:2459)
==47004==    by 0x4A07DB: process_input_packet (ffmpeg.c:2613)
==47004==
==47004== LEAK SUMMARY:
==47004==    definitely lost: 536 bytes in 1 blocks
==47004==    indirectly lost: 322,783 bytes in 5 blocks
==47004==      possibly lost: 0 bytes in 0 blocks
==47004==    still reachable: 32 bytes in 1 blocks
==47004==         suppressed: 0 bytes in 0 blocks
==47004== Reachable blocks (those to which a pointer was found) are not shown.
==47004== To see them, rerun with: --leak-check=full --show-leak-kinds=all
==47004==
==47004== For counts of detected and suppressed errors, rerun with: -v
==47004== ERROR SUMMARY: 1 errors from 1 contexts (suppressed: 0 from 0)
```

◀   ▶

Please confirm.
Thanks

Attachments (1)

- PoC.bmp(9.7 KB) - added by Suhwan 3 years ago.
  poc

Change History (2)

by Suhwan, 3 years ago

Attachment: *PoC.bmp*added

poc

comment:1 by Elon Musk, 3 years ago

Resolution: → fixed
Status: new → closed

Fixed in 673fce6d40d9a594fb7a0ea17d296b7d3d9ea856

Note: See TracTickets for help on using tickets.