

New issue

[Jump to bottom](#)

## A Division-By-Zero error in lib/extras/codec\_apng.cc jxl::DecodeImageAPNG() #308

Closed aug5t7 opened this issue on Jul 9, 2021 · 4 comments · Fixed by #313

Labels

cjxl

aug5t7 commented on Jul 9, 2021 • edited

## Describe the bug

A dividing by zero error in lib/extras/codec\_apng.cc:283 jxl::DecodeImageAPNG(), the `delay_den` can be 0 resulting in an Arithmetic Exception.

## To Reproduce

Steps to reproduce the behavior:

```
$ CC=clang CXX=clang++ CFLAGS="-g" CXXFLAGS="-g" cmake -DCMAKE_BUILD_TYPE=Release -DBUILD_TESTING=OFF ..
$ cmake --build . -- -j 8
$ tools/cjxl ./crash.png /tmp/tmp.jxl
```

The crash file [crash.png](#).

## Expected behavior

cjxl should encode the PNG to JXL successfully.

## Environment

- OS: 5.8.0-59-generic 20.04.1-Ubuntu
- Compiler version: clang version 7.0.1-12
- CPU type: x86\_64
- cjxl/djxl version string: cjxl v0.3.7 [AVX2,SSE4,Scalar]

## Additional context

```
# build without ASAN
$ ./cjxl ./crash.png /tmp/tmp.jxl
JPEG XL encoder v0.3.7 [AVX2,SSE4,Scalar]
[1] 1597419 floating point exception ./cjxl ./crash.png /tmp/tmp.jxl
```

eustas commented on Jul 9, 2021

Contributor

Thanks for the report. Will take a look soon.

eustas commented on Jul 9, 2021


Contributor

From stack-trace / report it looks like compiler / ASAN bug to me... Looking further.

eustas commented on Jul 9, 2021

Contributor

Can't reproduce.

 jonsneyers added the cjxl label on Jul 9, 2021

aug5t7 commented on Jul 9, 2021

Author

There seems to be a bug with ASAN... the report of ASAN is misleading ...Try to reproduce without ASAN?

The cause of this crash looks like a division by zero error in `lib/extras/codec_apng.cc:283 jxl::DecodeImageAPNG()`

```
...
280         if (hasInfo) {
281             if (!processing_finish(png_ptr, info_ptr)) {
282                 ImageBundle bundle(&io->metadata.m);
283                 bundle.duration = delay_num * 1000 / delay_den;
284                 bundle.origin.x0 = x0;
285                 bundle.origin.y0 = y0;
286             }
287         }
288     }
```

`delay_den` can be 0 according to [this wiki](#).

The `delay_num` and `delay_den` parameters together specify a fraction indicating the time to display the current frame, in seconds. If the denominator is 0, it is to be treated as if it were 100 (that is, `delay_num` then specifies 1/100ths of a second).

so the `delay_den` maybe need to be checked?

some debugging infomation from gdb

```
gdb-peda$ p delay_den
$2 = 0x0
```

```
gdb-peda$ n
Thread 1 "cjxl" received signal SIGFPE, Arithmetic exception.
[-----registers-----]
RAX: 0x3e8
RBX: 0x7fffffff5a8 --> 0x5555559d5170 --> 0x555555563ff0 (<jxl::Fields::~Fields()>: ret)
RCX: 0x0
RDX: 0x0
RSI: 0x0
RDI: 0x5555559fa750 --> 0x206c786a ('jxl ')
RBP: 0x7fffffff8e0 --> 0x7fffffff920 --> 0x7fffffffca10 --> 0x7fffffff30 --> 0x7ffffffe220 --> 0x0
RSP: 0x7fffffff590 --> 0x1
RIP: 0x5555558407f8 (<jxl::DecodeImageAPNG(jxl::Span<unsigned char const>, jxl::ThreadPool*, jxl::CodecInOut*)+2568>: div DWORD PTR [rbp-0x44])
R8 : 0x2
R9 : 0x2951 ('Q')
R10: 0x10ad83
R11: 0x10ad
R12: 0x5555559f34b0 --> 0x7ffffbad2488
R13: 0xc8
R14: 0x7fffffff340 --> 0xffffffffffffff
R15: 0xc8
EFLAGS: 0x10246 (carry PARITY adjust ZERO sign trap INTERRUPT direction overflow)
[-----code-----]
0x5555558407e8 <jxl::DecodeImageAPNG(jxl::Span<unsigned char const>, jxl::ThreadPool*, jxl::CodecInOut*)+2552>: movups XMMWORD PTR [rbx+0x150],xmm0
0x5555558407ef <jxl::DecodeImageAPNG(jxl::Span<unsigned char const>, jxl::ThreadPool*, jxl::CodecInOut*)+2559>: imul eax,DWORD PTR [rbp-0x40],0x3e8
0x5555558407f6 <jxl::DecodeImageAPNG(jxl::Span<unsigned char const>, jxl::ThreadPool*, jxl::CodecInOut*)+2566>: xor edx,edx
=> 0x5555558407f8 <jxl::DecodeImageAPNG(jxl::Span<unsigned char const>, jxl::ThreadPool*, jxl::CodecInOut*)+2568>: div DWORD PTR [rbp-0x44]
0x5555558407fb <jxl::DecodeImageAPNG(jxl::Span<unsigned char const>, jxl::ThreadPool*, jxl::CodecInOut*)+2571>: mov DWORD PTR [rbp-0x318],eax
0x555555840801 <jxl::DecodeImageAPNG(jxl::Span<unsigned char const>, jxl::ThreadPool*, jxl::CodecInOut*)+2577>: mov rcx,QWORD PTR [rbp-0x120]
0x555555840808 <jxl::DecodeImageAPNG(jxl::Span<unsigned char const>, jxl::ThreadPool*, jxl::CodecInOut*)+2584>: mov DWORD PTR [rbp-0x320],ecx
0x55555584080e <jxl::DecodeImageAPNG(jxl::Span<unsigned char const>, jxl::ThreadPool*, jxl::CodecInOut*)+2590>: mov rsi,QWORD PTR [rbp-0x118]
[-----stack-----]
0000| 0x7fffffff590 --> 0x1
0008| 0x7fffffff598 --> 0x0
0016| 0x7fffffff5a0 --> 0x1
0024| 0x7fffffff5a8 --> 0x5555559d5170 --> 0x555555563ff0 (<jxl::Fields::~Fields()>: ret)
0032| 0x7fffffff5b0 --> 0x0
0040| 0x7fffffff5b8 --> 0x0
0048| 0x7fffffff5c0 --> 0x0
0056| 0x7fffffff5c8 --> 0x0
[-----]
Legend: code, data, rodata, value
Stopped reason: SIGFPE
0x00005555558407f8 in jxl::DecodeImageAPNG (bytes=..., pool=<optimized out>, io=<optimized out>)
    at /home/au9/target/libjxl/libjxl/lib/extras/codec_apng.cc:283
283      bundle.duration = delay_num * 1000 / delay_den;
gdb-peda$ n
Couldn't get registers: No such process.
Couldn't get registers: No such process.
[Thread 0x7ffff5dfa700 (LWP 1792923) exited]
[Thread 0x7ffff65fb700 (LWP 1792921) exited]
[Thread 0x7ffff60fc700 (LWP 1792918) exited]
[Thread 0x7ffff75fd700 (LWP 1792917) exited]

Program terminated with signal SIGFPE, Arithmetic exception.
The program no longer exists.
```

I will re-edit the description of this issue :o)

 **aug5t7** changed the title ~~A stack use after scope issue with cjxl encode routine~~ **A Dividing-By-Zero error in lib/extras/codec\_apng.cc jxl::DecodeImageAPNG()** on Jul 9, 2021

 **aug5t7** changed the title ~~A Dividing-By-Zero error in lib/extras/codec\_apng.cc jxl::DecodeImageAPNG()~~ **A Division-By-Zero error in lib/extras/codec\_apng.cc jxl::DecodeImageAPNG()** on Jul 9, 2021

 **cagelight** mentioned this issue on Jul 10, 2021

**Fix handling of APNG with 0 delay\_den #313**

 Merged

 **deymo** closed this as completed in [#313](#) on Jul 15, 2021

Assignees

No one assigned

Labels

[cjxl](#)

Projects

None yet

Milestone

No milestone

Development

Successfully merging a pull request may close this issue.

 **Fix handling of APNG with 0 delay\_den**  
[cagelight/libjxl](#)

3 participants

