Exiv2 - Bug #1080

Division by zero / crash on malformed input file

13 May 2015 17:03 - Hanno Böck

Status:	Closed	**Start date:**	13 May 2015
Priority:	Normal	**Due date:**	
Assignee:	Robin Mills	**% Done:**	100%
Category:	metadata	**Estimated time:**	6.00 hours
Target version:	0.26		

Description

The attached file will cause a crash / integer division by zero in exiv2.

Backtrace:
#0 0x00007ffff7a6886f in Exiv2::Internal::print0x9204 (os=..., value=...) at tags.cpp:2551
#1 0x0000000000417b62 in Action::Print::printTag (this=this@entry=0x62e560, exifData=...,
key="Exif.Photo.ExposureBiasValue", label="Exposure bias") at actions.cpp:458
#2 0x0000000000418494 in Action::Print::printSummary (this=this@entry=0x62e560)
at actions.cpp:325
#3 0x000000000041a70c in Action::Print::run (this=0x62e560, path="exiv2-divzero.jpg")
at actions.cpp:236
#4 0x000000000040590e in main (argc=<optimized out>, argv=<optimized out>) at exiv2.cpp:171

This was found while fuzzing exiv2 with american fuzzy lop.

Related issues:

Related to Exiv2 - Bug #1129: Different behaviour of exiv2 between remote and...	**Closed**	**12 Oct 2015**

Associated revisions

Revision 4645 - 19 Oct 2016 19:06 - Robin Mills

#1080 Fix submitted.

Revision 4646 - 19 Oct 2016 19:18 - Robin Mills

#1080 Added the file BLANK.JPG to test suite as exiv2-bug1080.jpg

History

#1 - 08 Jun 2015 16:41 - Felix Bolte

- File 0001-fix_value_excess_of_max_int32t.patch added

hey, just because i stumpled over the same issue by using afl fuzzing, it is not the "/d" (division by zero) problem, it is the "std::abs(bias.first)" which fails, you can reproduce it on any image:

```
felix@between:~/afl-1.79b/trunk/build$ ./bin/exiv2 -p a /tmp/test.jpg
Exif.Image.Flash                             Short       1  (256)
felix@between:~/afl-1.79b/trunk/build$ ./bin/exiv2 -M"set Exif.Photo.ExposureBiasValue SRational -2147483647/1
" /tmp/test.jpg
felix@between:~/afl-1.79b/trunk/build$ ./bin/exiv2 -p a /tmp/test.jpg
Exif.Image.ExifTag                           Long        1  38
Exif.Photo.ExposureBiasValue                 SRational   1  -2147483647 EV
Exif.Image.Flash                             Short       1  (256)
felix@between:~/afl-1.79b/trunk/build$ ./bin/exiv2 -M"set Exif.Photo.ExposureBiasValue SRational -2147483648/1
" /tmp/test.jpg
felix@between:~/afl-1.79b/trunk/build$ ./bin/exiv2 -p a /tmp/test.jpg
Exif.Image.ExifTag                           Long        1  38
Exif.Photo.ExposureBiasValue                 SRational   1  --2147483648 EV
Exif.Image.Flash                             Short       1  (256)
felix@between:~/afl-1.79b/trunk/build$ ./bin/exiv2 -M"set Exif.Photo.ExposureBiasValue SRational -2147483648/1
3" /tmp/test.jpg
felix@between:~/afl-1.79b/trunk/build$ ./bin/exiv2 -p a /tmp/test.jpg
Exif.Image.ExifTag                           Long        1  38
Floating point exception (core dumped)
```

so problem is, that INT_MAX is 2147483647 but abs(-2147483648) is executed, i propose the attached patch to the current trunk, which also fixes the double minus in the output if EV is negative!

#2 - 09 Jun 2015 12:29 - Felix Bolte

i have to admit, that the above patch only works partly, if you compile with optimization (>= -O2), we still get a floating point exception on the bias values "-2147483648/12415" ... after some debugging, i found out, that gcd returns a correct "-1", but this number is somehow broken and not what we see, because abs(gcd)) still returns a -1, which is clearly wrong ... anyhow, values of "-2147483647/12415" do work, so it is still a max int problem somewhere ... after looking in the gcd template i see it is doing an own abs on the input, which might corrupt the outcome!?

```
    template <typename IntType>
    IntType gcd(IntType n, IntType m)
    {
        // Avoid repeated construction
        IntType zero(0);

        // This is abs() - given the existence of broken compilers with Koenig
        // lookup issues and other problems, I code this explicitly. (Remember,
        // IntType may be a user-defined type).
[...]
        if (n < zero)
            n = -n;
        if (m < zero)
            m = -m;
[...]

        // As n and m are now positive, we can be sure that %= returns a
        // positive value (the standard guarantees this for built-in types,
        // and we require it of user-defined types).
```

if i comment out the abs lines, the program works again even with gcc compiler optimization! ... reading the comments in the code makes clear why we need the abs inside the gcd function, but as you can see, this leads to another problem ...

so sorry, i guess i am stuck here and a more experienced exiv2 programmer has to look at this problem and fix the gcd construct :)

#3 - 21 Aug 2015 18:04 - Robin Mills

- Category set to metadata

- Status changed from New to Assigned

- Assignee set to Robin Mills

- Target version set to 0.26

#4 - 24 Sep 2015 11:09 - Robin Mills

- Estimated time set to 3.00 h

#5 - 10 Oct 2015 07:22 - Robin Mills

- % Done changed from 0 to 30

Felix

I'm unable to reproduce your report. The test file seems to be invalid.

```
650 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pS http://dev.exiv2.org/attachments/download/786/exiv2-divzero
.jpg
STRUCTURE OF JPEG FILE: http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg
 address | marker     | length  | data
       2 | 0xd8 SOI   |       0
       4 | 0xe1 APP1  |    4400 | Exif..MM.*.....00000000000000000
    4404 | 0xffffffff a?Z?Exiv2 exception in print action for file http://dev.exiv2.org/attachments/download/7
86/exiv2-divzero.jpg:
This does not look like a JPEG image
651 rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

If I attempt to run:

```
659 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pa http://dev.exiv2.org/attachments/download/786/exiv2-divzero
.jpg
Exiv2 exception in print action for file http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg:
Failed to read image data
660 rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

This is correct. image->readMetatdata() throws. So I never get near the gcd code.

Do you have your test file /tmp/test.jpg that breaks gcd()

#6 - 12 Oct 2015 16:48 - Felix Bolte

hi robin,

by coincidence you maybe found a small bug/feature, because when i download the attached image to the filesystem first, ill see the ticket error ... when using "-pa" on the URL i see this via strace:

```
recvfrom(3, 0x721f420d0320, 32768, 0, 0, 0) = -1 EAGAIN (Resource temporarily unavailable)
nanosleep({0, 0}, 0x721f420d0260)       = 0
recvfrom(3, "HTTP/1.1 200 OK\r\nDate: Mon, 12 O"..., 32768, 0, NULL, NULL) = 4877
recvfrom(3, 0x721f420d0320, 32768, 0, 0, 0) = -1 EAGAIN (Resource temporarily unavailable)
nanosleep({0, 0}, 0x721f420d0260)       = 0
recvfrom(3, "", 32768, 0, NULL, NULL)   = 0
close(-1)                               = -1 EBADF (Bad file descriptor)
close(3)                                = 0
write(2, "Exiv2 exception in print action "..., 41Exiv2 exception in print action for file ) = 41
write(2, "http://dev.exiv2.org/attachments"..., 63http://dev.exiv2.org/attachments/download/786/exiv2-divzero.
jpg) = 63
write(2, ":\n", 2:
)                       = 2
write(2, "Failed to read image data", 25Failed to read image data) = 25
write(2, "\n", 1
)                        = 1
exit_group(1)                           = ?
+++ exited with 1 +++
```

#7 - 12 Oct 2015 18:01 - Robin Mills

Interesting find, Felix and you are partly correct.

```
rmills@rmillsmbp:~/gnu/exiv2/trunk $ curl -O http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  4404    0  4404    0     0  62061      0 --:--:-- --:--:-- --:--:-- 64764
rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

Option -pS reports the same using local and remote I/O.

FileIo:

```
rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pS ./exiv2-divzero.jpg
STRUCTURE OF JPEG FILE: ./exiv2-divzero.jpg
 address | marker     | length  | data
       2 | 0xd8 SOI   |       0
       4 | 0xe1 APP1  |    4400 | Exif..MM.*.....00000000000000000
    4404 | 0xffffffff ?? Exiv2 exception in print action for file ./exiv2-divzero.jpg:
This does not look like a JPEG image
rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

HttpIo:

```
rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pS http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg
STRUCTURE OF JPEG FILE: http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg
 address | marker     | length  | data
       2 | 0xd8 SOI   |       0
       4 | 0xe1 APP1  |    4400 | Exif..MM.*.....00000000000000000
    4404 | 0xffffffff ?+ Exiv2 exception in print action for file http://dev.exiv2.org/attachments/download/78
6/exiv2-divzero.jpg:
This does not look like a JPEG image
```

However option -pa behaves differently and, as you have reported, does ultimately throw a floating point exception when the file is local. However it fails gracefully with httpio.

FileIo:

```
rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pa exiv2-divzero.jpg
Error: Directory Image: Next pointer is out of bounds; ignored.
Warning: Directory Image, entry 0x3030 has unknown Exif (TIFF) type 12336; setting type size 1.
Error: Directory Image, entry 0x3030 has invalid size 808464432*1; skipping entry.
Warning: Directory Image, entry 0x3030 has unknown Exif (TIFF) type 12336; setting type size 1.
... deleted about 100 more reports of this ...
```

```
Warning: JPEG format error, rc = 5
Exif.Image.ExifTag                            Long        1  217
Floating point exception: 8
rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

Httplo:

```
rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pa http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg
Exiv2 exception in print action for file http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg:
Failed to read image data
rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

This is a step forward. It looks to me as though the local file version has ignored the message:

```
Warning: JPEG format error, rc = 5
```

This is surprising as our I/O code presents a Basiclo interface and hides the source from the client who in this case is jpgimage.cpp.

I'll step the code in the debugger to see what's going on here. I feel the resolution of this will be in FileIo and not gcd().

Incidentally, we have two implementations of class HttpIo. We use libcurl with you ./configure --enable-webready. We have a "no thrills" default http implementation to ensure that http is supported on all builds of exiv2 v0.25 and later. I've built both ways and HttpIo is consistent in saying "Failed to read image data".

#8 - 12 Oct 2015 18:57 - Felix Bolte

- File Blank.JPG added

> This is a step forward. It looks to me as though the local file version has ignored the message:
>
> ```
> Warning: JPEG format error, rc = 5
> ```

just to clarify, you can easily take any image and add the "bad" exif data yourself (see new attachment), without any further error:

```
felix@toaster:~/old_afl-1.80b/exiv2-trunk/build$ ./bin/exiv2 -pS /tmp/Blank.JPG
STRUCTURE OF JPEG FILE: /tmp/Blank.JPG
 address | marker     | length  | data
       2 | 0xd8 SOI   |       0
       4 | 0xe0 APP0  |      16 | JFIF.....`.`.....C.............
      22 | 0xdb DQT   |      67
      91 | 0xdb DQT   |      67
     160 | 0xc0 SOF0  |      17
     179 | 0xc4 DHT   |      31
     212 | 0xc4 DHT   |     181
     395 | 0xc4 DHT   |      31
     428 | 0xc4 DHT   |     181
     611 | 0xda SOS   |      12
felix@toaster:~/old_afl-1.80b/exiv2-trunk/build$ ./bin/exiv2 -pa /tmp/Blank.JPG
felix@toaster:~/old_afl-1.80b/exiv2-trunk/build$ ./bin/exiv2 -M"set Exif.Photo.ExposureBiasValue SRational -21
47483648/13" /tmp/Blank.JPG
felix@toaster:~/old_afl-1.80b/exiv2-trunk/build$ ./bin/exiv2 -pS /tmp/Blank.JPG
STRUCTURE OF JPEG FILE: /tmp/Blank.JPG
 address | marker     | length  | data
       2 | 0xd8 SOI   |       0
       4 | 0xe0 APP0  |      16 | JFIF.....`.`.....<Exif..II*....
      22 | 0xe1 APP1  |      60 | Exif..II*.......i..............
      84 | 0xdb DQT   |      67
     153 | 0xdb DQT   |      67
     222 | 0xc0 SOF0  |      17
     241 | 0xc4 DHT   |      31
     274 | 0xc4 DHT   |     181
     457 | 0xc4 DHT   |      31
     490 | 0xc4 DHT   |     181
     673 | 0xda SOS   |      12
felix@toaster:~/old_afl-1.80b/exiv2-trunk/build$ ./bin/exiv2 -pa /tmp/Blank.JPG
Exif.Image.ExifTag                            Long        1  26
Floating point exception
```

#9 - 12 Oct 2015 20:07 - Robin Mills

- Assignee changed from Robin Mills to Andreas Huggel

- % Done changed from 30 to 50

- Estimated time changed from 3.00 h to 5.00 h

Thanks. Your file Blank.JPG reproduces this fault.

I believe ExposureBiasValue is normally a small number to over/under expose the image. 1.0 = Normal

```
rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pa -g Bias ~/Pictures/2015/Roof/DSC_7443.jpg
Exif.Photo.ExposureBiasValue                 SRational   1  0 EV
Exif.Nikon3.WhiteBalanceBias                 SShort      2  0 0
rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

In this case you've set to an astronomical negative number. Exiv2 is not a metadata policeman. We give you the tools and you can set fields to any value even if they don't make sense.

None the less, exiv2 should not crash. One simple fix is:

```
    std::ostream& print0x9204(std::ostream& os, const Value& value, const ExifData*)
    {
        Rational bias = value.toRational();
        if (bias.first == 0) {
            os << "0 EV";
        } else {
            double  d = (double)bias.first / (double)bias.second ;
            os << "(" << bias.first << "/" << bias.second << ") = " << d << " EV";
        }
        return os;
    }
```

And results in the output:

```
rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pa http://dev.exiv2.org/attachments/download/849/Blank.JPG
Exif.Image.ExifTag                           Long        1  26
Exif.Photo.ExposureBiasValue                 SRational   1  (-2147483648/13) = -1.65191e+08 EV
```

I didn't write this code. Without knowing why the code jumps through those hoops, I'm reluctant to change it. Running the code through svn blame reveals that Andreas added this a long time ago. I'm going to assign this to him for comment:

```
rmills@rmillsmbp:~/gnu/exiv2/trunk $ svn blame src/tags.cpp | nl -ba | head -2592 | tail -22
  2571      1512    ahuggel     std::ostream& print0x9204(std::ostream& os, const Value& value, const ExifData
*)
  2572       169    ahuggel     {
  2573       169    ahuggel         Rational bias = value.toRational();
  2574       181    ahuggel         if (bias.second <= 0) {
  2575       181    ahuggel             os << "(" << bias.first << "/" << bias.second << ")";
  2576       169    ahuggel         }
  2577       181    ahuggel         else if (bias.first == 0) {
  2578      1720    ahuggel             os << "0 EV";
  2579       181    ahuggel         }
  2580       169    ahuggel         else {
  2581       616    ahuggel             int32_t d = gcd(bias.first, bias.second);
  2582       616    ahuggel             int32_t num = std::abs(bias.first) / d;
  2583       616    ahuggel             int32_t den = bias.second / d;
  2584       181    ahuggel             os << (bias.first < 0 ? "-" : "+") << num;
  2585       181    ahuggel             if (den != 1) {
  2586       181    ahuggel                 os << "/" << den;
  2587       181    ahuggel             }
  2588      1720    ahuggel             os << " EV";
  2589       169    ahuggel         }
  2590       169    ahuggel         return os;
  2591       169    ahuggel     }
  2592       169    ahuggel
rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

I've opened a new issue #1129 to investigate the FileIo/HttpIo anomaly observed on exiv2-divzero.jpg

#10 - 13 Oct 2015 16:12 - Robin Mills

- File img_4381.jpg added

While I was clearing the leaves in the garden today, I started thinking about ExposureBiasValue. So, I got the little Canon camera and took a series of photos using the the exposure +/- control. Here's the metadata:

```
722 rmills@rmillsmbp:~/Pictures/2015/October $ exiv2 -pa --grep ExposureBias img_43{75..87}.jpg
img_4375.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  -2 EV
img_4376.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  -5/3 EV
img_4377.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  -4/3 EV
img_4378.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  -1 EV
img_4379.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  -2/3 EV
img_4380.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  -1/3 EV
img_4381.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  0 EV
img_4382.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  +1/3 EV
img_4383.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  +2/3 EV
img_4384.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  +1 EV
img_4385.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  +4/3 EV
img_4386.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  +5/3 EV
img_4387.jpg          Exif.Photo.ExposureBiasValue                 SRational   1  +2 EV
723 rmills@rmillsmbp:~/Pictures/2015/October $
```

I believe the EV is the increase in F stops. So EV = +1 indicates that the lens has been opened by one F stop. So an EV of -10+8 would indicate that the lens has been closed by 100 million F stops. That's a lot of F stops!

img_4381_small.jpg

#11 - 13 Oct 2015 16:22 - Robin Mills

- File img_4381_small.jpg added

#12 - 16 Sep 2016 07:19 - Robin Mills

- Status changed from Assigned to Closed

- % Done changed from 50 to 100

I believe this is closed.

```
797 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pa --grep exposure/i http://dev.exiv2.org/attachments/download
/851/img_4381_small.jpg
Exif.Photo.ExposureTime                      Rational    1  1/40 s
Exif.Photo.ExposureBiasValue                 SRational   1  0 EV
Exif.CanonCs.ExposureProgram                 Short       1  Program (P)
Exif.Photo.ExposureMode                      Short       1  Auto
798 rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

#13 - 16 Sep 2016 09:33 - Felix Bolte

please try with the BLANK.jpg i uploaded to this this ticket ... still broken here:

```
[felix@between trunk]$ svn info
Path: .
Working Copy Root Path: /home/felix/Downloads/exiv2/trunk
URL: svn://dev.exiv2.org/svn/trunk
Relative URL: ^/trunk
Repository Root: svn://dev.exiv2.org/svn
Repository UUID: b7c8b350-86e7-0310-a4b4-de8f6a8f16a3
Revision: 4501
Node Kind: directory
Schedule: normal
Last Changed Author: robinwmills
Last Changed Rev: 4501
Last Changed Date: 2016-09-16 07:33:40 +0200 (Fri, 16 Sep 2016)

[felix@between trunk]$ ./bin/exiv2 -pa /tmp/Blank.JPG
Exif.Image.ExifTag                           Long        1  26
Floating point exception (core dumped)
```

#14 - 16 Sep 2016 10:12 - Robin Mills

- Status changed from Closed to Assigned

- % Done changed from 100 to 50

Yes. I was too enthusiastic to close this. I'm putting this back to 50% done and asking Andreas to investigate.

#15 - 19 Oct 2016 19:07 - Robin Mills

- Status changed from Assigned to Closed

- Assignee changed from Andreas Huggel to Robin Mills

- % Done changed from 50 to 100

- Estimated time changed from 5.00 h to 6.00 h

Fix submitted r4645

```
796 rmills@rmillsmbp:~/gnu/exiv2/trunk/build $ bin/Debug/exiv2 -pR http://dev.exiv2.org/attachments/download/8
49/Blank.JPG
STRUCTURE OF JPEG FILE: http://dev.exiv2.org/attachments/download/849/Blank.JPG
 address | marker       |  length | data
       0 | 0xffd8 SOI
       2 | 0xffe0 APP0  |      16 | JFIF.....`.`....
      20 | 0xffe1 APP1  |      60 | Exif..II*.......i.............
  STRUCTURE OF TIFF FILE (II): MemIo
   address |    tag                          |      type |    count |    offset | value
        10 | 0x8769 ExifTag                  |      LONG |        1 |        26 | 26
    STRUCTURE OF TIFF FILE (II): MemIo
     address |    tag                            |      type |    count |    offset | value
          28 | 0x9204 ExposureBiasValue          | SRATIONAL |        1 |        44 | 44/0
    END MemIo
  END MemIo
     82 | 0xffdb DQT   |      67
    151 | 0xffdb DQT   |      67
    220 | 0xffc0 SOF0  |      17
    239 | 0xffc4 DHT   |      31
    272 | 0xffc4 DHT   |     181
    455 | 0xffc4 DHT   |      31
    488 | 0xffc4 DHT   |     181
    671 | 0xffda SOS
797 rmills@rmillsmbp:~/gnu/exiv2/trunk/build $ bin/Debug/exiv2 -pa http://dev.exiv2.org/attachments/download/8
49/Blank.JPG
Exif.Image.ExifTag                            Long        1  26
Exif.Photo.ExposureBiasValue                  SRational   1  0 EV
798 rmills@rmillsmbp:~/gnu/exiv2/trunk/build $
```

Files

exiv2-divzero.jpg	4.3 KB	13 May 2015	Hanno Böck
0001-fix_value_excess_of_max_int32t.patch	773 Bytes	08 Jun 2015	Felix Bolte
Blank.JPG	693 Bytes	12 Oct 2015	Felix Bolte
img_4381.jpg	4.89 MB	13 Oct 2015	Robin Mills
img_4381_small.jpg	252 KB	13 Oct 2015	Robin Mills