Closed Issue created 5 months ago by Marsman

tiffsplit: stack-buffer-overflow in _TIFFVGetField() (CVE-2022-34526)

Summary

A stack overflow occurs when tiffsplit processes a craft file.

Version

LIBTIFF, Version 4.4.0, master b2d61984

Steps to reproduce

Download the poc file from here and run cmd `$ tiffsplit $POC`

Platform

Ubuntu 16.04.3 LTS, x86_64, Clang-12

ASAN report

```
$ ./bin_asan/bin/tiffsplit ./poc-tiffsplit-b2d61984-_TIFFVGetField-stackoverflow
TIFFReadDirectoryCheckOrder: Warning, Invalid TIFF directory; tags are not sorted in ascending order
TIFFReadDirectory: Warning, Unknown field with tag 317 (0x13d) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 34893 (0x884d) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 31350 (0x7a76) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 59310 (0xe7ae) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 520 (0x208) encountered.
TIFFFetchNormalTag: Warning, IO error during reading of "DocumentName"; tag ignored.
TIFFFetchNormalTag: Warning, Incorrect count for "PhotometricInterpretation"; tag ignored.
./poc-tiffsplit-b2d61984-_TIFFVGetField-stackoverflow: ZSTD compression support is not configured.
=================================================================
==21665==ERROR: AddressSanitizer: stack-buffer-overflow on address 0x7ffcf0f98d50 at pc 0x0000004ddb
WRITE of size 4 at 0x7ffcf0f98d50 thread T0
    #0 0x4ddb2c in _TIFFVGetField /opt/disk/marsman/libtiff/b2d61984/build_asan/libtiff/../../code/l
    #1 0x4cd95e in TIFFVGetField /opt/disk/marsman/libtiff/b2d61984/build_asan/libtiff/../../code/li
    #2 0x4cd95e in TIFFGetField /opt/disk/marsman/libtiff/b2d61984/build_asan/libtiff/../../code/lib
    #3 0x4c9c58 in tiffcp /opt/disk/marsman/libtiff/b2d61984/build_asan/tools/../../code/tools/tiffs
    #4 0x4c9c58 in main /opt/disk/marsman/libtiff/b2d61984/build_asan/tools/../../code/tools/tiffspl
    #5 0x7f221a17f83f in __libc_start_main /build/glibc-S7Ft5T/glibc-2.23/csu/../csu/libc-start.c:29
    #6 0x41ce88 in _start (/opt/disk/marsman/libtiff/b2d61984/bin_asan/bin/tiffsplit+0x41ce88)

Address 0x7ffcf0f98d50 is located in stack of thread T0 at offset 144 in frame
    #0 0x4c8fbf in main /opt/disk/marsman/libtiff/b2d61984/build_asan/tools/../../code/tools/tiffspl

  This frame has 18 object(s):
    [32, 40) 'bytecounts.i371.i' (line 316)
    [64, 72) 'bytecounts.i.i' (line 354)
    [96, 98) 'bitspersample.i' (line 237)
    [112, 114) 'samplesperpixel.i' (line 237)
    [128, 130) 'compression.i' (line 237)
    [144, 146) 'shortv.i' (line 237) <== Memory access at offset 144 partially overflows this variab
    [160, 168) 'shortav.i' (line 237)
    [192, 196) 'w.i' (line 238)
    [208, 212) 'l.i' (line 238)
    [224, 228) 'floatv.i' (line 239)
    [240, 248) 'stringv.i' (line 240)
    [272, 276) 'longv.i' (line 241)
    [288, 292) 'count.i' (line 252)
    [304, 312) 'table.i' (line 253)
    [336, 344) 'red.i' (line 280)
    [368, 376) 'green.i' (line 280)
    [400, 408) 'blue.i' (line 280)
    [432, 434) 'shortv2.i' (line 284)
HINT: this may be a false positive if your program uses some custom stack unwind mechanism, swapcont
      (longjmp and C++ exceptions *are* supported)
```

```
SUMMARY: AddressSanitizer: stack-buffer-overflow /opt/disk/marsman/libtiff/b2d61984/build_asan/libti
Shadow bytes around the buggy address:
  0x10001e1eb150: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10001e1eb160: 00 00 00 00 f1 f1 f1 f1 00 00 00 f3 f3 f3 f3 f3
  0x10001e1eb170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10001e1eb180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10001e1eb190: 00 00 00 00 00 00 00 00 f1 f1 f1 f1 f8 f2 f2 f2
=>0x10001e1eb1a0: f8 f2 f2 f2 02 f2 02 f2 02 f2[02]f2 00 f2 f2 f2
  0x10001e1eb1b0: 04 f2 04 f2 04 f2 00 f2 f2 f2 04 f2 f8 f2 f8 f2
  0x10001e1eb1c0: f2 f2 f8 f2 f2 f2 f8 f2 f2 f2 f8 f2 f2 f2 f8 f3
  0x10001e1eb1d0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10001e1eb1e0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10001e1eb1f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
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
  Shadow gap:              cc
==21665==ABORTING
```

GDB report

```
Starting program: /opt/disk/marsman/libtiff/b2d61984/bin_normal/bin/tiffsplit ./poc-tiffsplit-b2d619
[Thread debugging using libthread_db enabled]
Using host libthread_db library "/lib/x86_64-linux-gnu/libthread_db.so.1".
TIFFReadDirectoryCheckOrder: Warning, Invalid TIFF directory; tags are not sorted in ascending order
TIFFReadDirectory: Warning, Unknown field with tag 317 (0x13d) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 34893 (0x884d) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 31350 (0x7a76) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 59310 (0xe7ae) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 520 (0x208) encountered.
TIFFFetchNormalTag: Warning, IO error during reading of "DocumentName"; tag ignored.
TIFFFetchNormalTag: Warning, Incorrect count for "PhotometricInterpretation"; tag ignored.
./poc-tiffsplit-b2d61984-_TIFFVGetField-stackoverflow: ZSTD compression support is not configured.

Program received signal SIGSEGV, Segmentation fault.
_TIFFVGetField (tif=<optimized out>, tag=<optimized out>, ap=0x7fffffffdaa0) at ../../code/libtiff/t
1167                                                *va_arg(ap, const void **) = tv->value;
(gdb) bt
#0  _TIFFVGetField (tif=<optimized out>, tag=<optimized out>, ap=0x7fffffffdaa0) at ../../code/libti
#1  0x0000000000407dc4 in TIFFGetField (tif=tif@entry=0x671010, tag=tag@entry=317) at ../../code/lib
#2  0x0000000000402f76 in tiffcp (out=0x6722e0, in=0x671010) at ../../code/tools/tiffsplit.c:260
#3  main (argc=<optimized out>, argv=<optimized out>) at ../../code/tools/tiffsplit.c:160
```

Drag your designs here or click to upload.

Tasks 0

No tasks are currently assigned. Use tasks to break down this issue into smaller parts.

Linked items 0

Link issues together to show that they're related or that one is blocking others. Learn more.

Related merge requests 1

_TIFFCheckFieldIsValidForCodec(): return FALSE when passed a codec-specific...

!363

When this merge request is accepted, this issue will be closed automatically.

Activity

Nikola Forró @nforro · 4 months ago — Contributor

What happens here is that libtiff is compiled without ZSTD compression support, but `_TIFFCheckFieldIsValidForCodec()` still returns true for `TIFFTAG_PREDICTOR` (should it?), so a generic `_TIFFVGetField()` is used to read the tag and it crashes (not exactly sure why).

Even Rouault mentioned in merge request !363 (merged) 4 months ago

Even Rouault mentioned in commit 5e60251e 4 months ago

Even Rouault closed via commit 275735d0 4 months ago

Even Rouault mentioned in commit Su_Laus/test@275735d0 4 months ago

Even Rouault mentioned in issue #486 (closed) 1 week ago

Even Rouault changed title from **tiffsplit: stack-buffer-overflow in _TIFFVGetField()** to **tiffsplit: stack-buffer-overflow in _TIFFVGetField() (CVE-2022-34526)** 1 week ago

Please register or sign in to reply