Closed Issue created 8 months ago by 4ugustus

tiffcrop: heap-buffer-overflow in writeSingleSection, tiffcrop.c:7345

Summary

There is a heap-buffer-overflow in writeSingleSection in tools/tiffcrop.c:7345. Remote attackers could leverage this vulnerability to cause a denial-of-service via a crafted tiff file.

Version

LIBTIFF, Version 4.3.0, commit id 5e180045 (Fri Feb 25 10:38:31 2022 +0000)

Steps to reproduce

```
# CFLAGS="-g -fsanitize=address -fno-omit-frame-pointer" CXXFLAGS="-g -fsanitize=address -fno-omit-f

# make -j; make install; make clean

./build_asan/bin/tiffcrop -H 341 poc /tmp/foo
TIFFReadDirectoryCheckOrder: Warning, Invalid TIFF directory; tags are not sorted in ascending order
TIFFReadDirectory: Warning, Unknown field with tag 501 (0x1f5) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 292 (0x124) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 18761 (0x4949) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 302 (0x12e) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 30692 (0x77e4) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 123 (0x7b) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 20932 (0x51c4) encountered.
TIFFFetchNormalTag: Warning, ASCII value for tag "InkNames" does not end in null byte. Forcing it to
TIFFFetchNormalTag: Warning, IO error during reading of "Tag 501"; tag ignored.
TIFFFetchNormalTag: Warning, Incompatible type for "Orientation"; tag ignored.
TIFFFetchNormalTag: Warning, Incorrect count for "ResolutionUnit"; tag ignored.
TIFFFetchNormalTag: Warning, IO error during reading of "Tag 20932"; tag ignored.
TIFFAdvanceDirectory: Error fetching directory count.
loadImage: Image lacks Photometric interpretation tag.
Fax4Decode: Bad code word at line 0 of strip 0 (x 58).
Fax4Decode: Warning, Premature EOL at line 0 of strip 0 (got 58, expected 12293).
Fax4Decode: Uncompressed data (not supported) at line 1 of strip 0 (x 57).
Fax4Decode: Warning, Premature EOL at line 1 of strip 0 (got 57, expected 12293).
Fax4Decode: Warning, Line length mismatch at line 22 of strip 0 (got 12296, expected 12293).
Fax4Decode: Warning, Premature EOL at line 23 of strip 0 (got 12288, expected 12293).
=================================================================
==820417==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x602000000091 at pc 0x7ffff75e8a
READ of size 1 at 0x602000000091 thread T0
    #0 0x7ffff75e8a6c  (/lib/x86_64-linux-gnu/libasan.so.5+0x67a6c)
    #1 0x5555555a69a5 in writeSingleSection /home/data/wdw/programs/libtiff/tools/tiffcrop.c:7345
    #2 0x5555555a5bac in writeImageSections /home/data/wdw/programs/libtiff/tools/tiffcrop.c:7110
    #3 0x55555558bb74 in main /home/data/wdw/programs/libtiff/tools/tiffcrop.c:2451
    #4 0x7ffff6d870b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x240b2)
    #5 0x55555558260d in _start (/home/data/wdw/programs/libtiff/build_asan/bin/tiffcrop+0x2e60d)

0x602000000091 is located 0 bytes to the right of 1-byte region [0x602000000090,0x602000000091)
allocated by thread T0 here:
    #0 0x7ffff768ebc8 in malloc (/lib/x86_64-linux-gnu/libasan.so.5+0x10dbc8)
    #1 0x55555561dc89 in _TIFFmalloc /home/data/wdw/programs/libtiff/libtiff/tif_unix.c:314
    #2 0x5555555b3b41 in setByteArray /home/data/wdw/programs/libtiff/libtiff/tif_dir.c:52
    #3 0x5555555b3c53 in _TIFFsetNString /home/data/wdw/programs/libtiff/libtiff/tif_dir.c:62
    #4 0x5555555ba2b9 in _TIFFVSetField /home/data/wdw/programs/libtiff/libtiff/tif_dir.c:485
    #5 0x5555556429d6 in Fax3VSetField /home/data/wdw/programs/libtiff/libtiff/tif_fax3.c:1205
    #6 0x5555555be292 in TIFFVSetField /home/data/wdw/programs/libtiff/libtiff/tif_dir.c:890
    #7 0x5555555bdca2 in TIFFSetField /home/data/wdw/programs/libtiff/libtiff/tif_dir.c:834
    #8 0x5555555ec6a7 in TIFFFetchNormalTag /home/data/wdw/programs/libtiff/libtiff/tif_dirread.c:53
    #9 0x5555555e25cf in TIFFReadDirectory /home/data/wdw/programs/libtiff/libtiff/tif_dirread.c:400
    #10 0x555555609554 in TIFFClientOpen /home/data/wdw/programs/libtiff/libtiff/tif_open.c:484
    #11 0x55555561da04 in TIFFFdOpen /home/data/wdw/programs/libtiff/libtiff/tif_unix.c:209
    #12 0x55555561dc44 in TIFFOpen /home/data/wdw/programs/libtiff/libtiff/tif_unix.c:248
    #13 0x55555558a8f1 in main /home/data/wdw/programs/libtiff/tools/tiffcrop.c:2261
    #14 0x7ffff6d870b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x240b2)
```

```
SUMMARY: AddressSanitizer: heap-buffer-overflow (/lib/x86_64-linux-gnu/libasan.so.5+0x67a6c)
Shadow bytes around the buggy address:
  0x0c047fff7fc0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7fd0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7fe0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff8000: fa fa 00 00 fa fa 02 fa fa fa fd fa fa fa fd fa
=>0x0c047fff8010: fa fa[01]fa fa fa fd fa fa fa fd fa fa fa 02 fa
  0x0c047fff8020: fa fa fd fa fa fa 00 fa fa fa fd fa fa fa 00 fa
  0x0c047fff8030: fa fa 02 fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8040: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8050: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8060: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
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
==820417==ABORTING
```

Platform

```
# uname -a
Linux 4a409ce47130 5.4.0-70-generic #78~18.04.1-Ubuntu SMP Sat Mar 20 14:10:07 UTC 2021 x86_64 x86_6
```

poc

Edited 8 months ago by 4ugustus

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 2

fix heap-buffer-overflow in tiffcp and tiffcrop (#277 and #398)
!313

Revised handling of TIFFTAG_INKNAMES and related TIFFTAG_NUMBEROFINKS value (fixes #149, #150, #1...
!385

When these merge requests are accepted, this issue will be closed automatically.

Activity

4ugustus changed the description 8 months ago ·

4ugustus mentioned in issue #277 8 months ago

4ugustus @waugustus · 8 months ago Author Contributor

Analysis

Crash cause

This crash is very similar to the heap-buffer-overflow in tiffcp.c:948, which is described in #277.

To trigger this crash, `page.mode != PAGE_MODE_NONE` needs to be set in tiffcrop.c:2410

```
if (page.mode == PAGE_MODE_NONE)
```

From the code, we can see that page.mode will be changed if at least one of the following parameters is present: `["-H", "-J", "-K", "-P", "-S", "-V"]`

Other analysis is the same as #277.

How to fix (No idea)

As described in #277, ~~I have no idea to check the bounds of *cp* in tiffcp right now, and may need your help.~~

```
uint16_t ninks;
const char* inknames;
uint16_t samplesperpixel = image->spp; // avoid to change the value of spp
if (TIFFGetField(in, TIFFTAG_NUMBEROFINKS, &ninks)) {
    TIFFSetField(out, TIFFTAG_NUMBEROFINKS, ninks);
    if (TIFFGetField(in, TIFFTAG_INKNAMES, &inknames)) {
        int inknameslen = strlen(inknames) + 1;
        const char* cp = inknames;
        while (samplesperpixel > 1) {
            cp = strchr(cp, '\0');
            if (cp) {
                cp++;
                inknameslen += (strlen(cp) + 1);
            }
            samplesperpixel--;
        }
        TIFFSetField(out, TIFFTAG_INKNAMES, inknameslen, inknames);
    }
}
```

Note that this bug also exists in writeCroppedImage, tiffcrop.c:8025 and needs to be fixed together.

Edited by 4ugustus 8 months ago

4ugustus mentioned in merge request !313 8 months ago

Su Laus @Su_Laus · 2 months ago Developer

The source of the error seems to be the same as described in #269 (comment 1079739024).

4ugustus @waugustus · 2 months ago Author Contributor

Thanks for your response. I agree that this bug is similar to #269 (closed). I haven't figured out a proper way to fix these without modifying the function interface. I try to replace `nink` with `spp`, as described in #277 and !313 . Hope this helps.

Su Laus mentioned in merge request !385 (merged) 2 months ago

Even Rouault closed via merge request !385 (merged) 1 month ago

Su Laus mentioned in commit f00484b9 1 month ago

Even Rouault mentioned in commit e8131125 1 month ago