Closed Issue created 9 months ago by 4ugustus

tiffcp: Assertion failed in TIFFReadAndRealloc, tif_read.c:99

Summary

There is a reachable assertion-failed crash in _TIFFReadAndRealloc, tif_read.c:99. Remote attackers could leverage this vulnerability to cause a denial-of-service via a crafted tiff file. **Note that this crash is different from #377 (closed)**

Version

573e0252 (Sun Feb 20 14:47:49 2022 +0100)

Steps to reproduce

```
$ tiffcp poc /tmp/foo

TIFFReadDirectoryCheckOrder: Warning, Invalid TIFF directory; tags are not sorted in ascending order
TIFFReadDirectory: Warning, Unknown field with tag 65535 (0xffff) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 65046 (0xfe16) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 53693 (0xd1bd) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 2449 (0x991) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 52970 (0xceea) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 3 (0x3) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 1 (0x1) encountered.
TIFFFetchNormalTag: Warning, ASCII value for tag "Model" does not end in null byte.
TIFFFetchNormalTag: Warning, Incorrect count for "FillOrder"; tag ignored.
TIFFFetchNormalTag: Warning, ASCII value for tag "DocumentName" contains null byte in value; value i
TIFFFetchNormalTag: Warning, ASCII value for tag "Tag 65046" does not end in null byte. Forcing it t
TIFFFetchNormalTag: Warning, Incorrect count for "XResolution"; tag ignored.
Fax4Decode: Warning, Line length mismatch at line 1 of strip 0 (got 60704, expected 60703).
Fax4Decode: Warning, Line length mismatch at line 3 of strip 0 (got 60704, expected 60703).
Fax4Decode: Bad code word at line 6 of strip 0 (x 6).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 6, expected 60703).
Fax4Decode: Bad code word at line 6 of strip 0 (x 0).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 0, expected 60703).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 8, expected 60703).
Fax4Decode: Uncompressed data (not supported) at line 6 of strip 0 (x 0).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 0, expected 60703).
Fax4Decode: Uncompressed data (not supported) at line 6 of strip 0 (x 60700).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 60700, expected 60703).
Fax4Decode: Uncompressed data (not supported) at line 6 of strip 0 (x 60700).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 60700, expected 60703).
Fax4Decode: Warning, Line length mismatch at line 6 of strip 0 (got 60704, expected 60703).
Fax4Decode: Warning, Premature EOF at line 6 of strip 0 (x 54).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 54, expected 60703).
Fax4Decode: Warning, Premature EOF at line 6 of strip 0 (x 0).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 0, expected 60703).
Fax4Decode: Warning, Premature EOF at line 6 of strip 0 (x 0).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 0, expected 60703).
Fax4Decode: Warning, Premature EOF at line 6 of strip 0 (x 0).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 0, expected 60703).
Fax4Decode: Warning, Premature EOF at line 6 of strip 0 (x 0).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 0, expected 60703).
Fax4Decode: Warning, Premature EOF at line 6 of strip 0 (x 0).
Fax4Decode: Warning, Premature EOL at line 6 of strip 0 (got 0, expected 60703).
TIFFReadDirectoryCheckOrder: Warning, Invalid TIFF directory; tags are not sorted in ascending order
TIFFReadDirectory: Warning, Unknown field with tag 4 (0x4) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 3 (0x3) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 5 (0x5) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 55941 (0xda85) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 51248 (0xc830) 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 65535 (0xffff) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 436 (0x1b4) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 64790 (0xfd16) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 2048 (0x800) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 6010 (0x177a) encountered.
```

```
TIFFReadDirectory: Warning, Unknown field with tag 60138 (0xeaea) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 16384 (0x4000) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 59904 (0xea00) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 8832 (0x2280) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 24655 (0x604f) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 62085 (0xf285) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 59152 (0xe710) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 27651 (0x6c03) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 392 (0x188) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 769 (0x301) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 0 (0x0) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 38573 (0x96ad) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 60159 (0xeaff) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 6144 (0x1800) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 12076 (0x2f2c) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 5327 (0x14cf) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 8289 (0x2061) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 34828 (0x880c) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 31820 (0x7c4c) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 62632 (0xf4a8) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 12006 (0x2ee6) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 50183 (0xc407) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 3840 (0xf00) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 16 (0x10) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 31365 (0x7a85) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 252 (0xfc) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 30069 (0x7575) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 18763 (0x494b) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 3505 (0xdb1) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 1 (0x1) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 9 (0x9) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 1002 (0x3ea) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 770 (0x302) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 59925 (0xea15) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 18761 (0x4949) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 2 (0x2) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 32768 (0x8000) encountered.
TIFFReadDirectory: Warning, Unknown field with tag 58339 (0xe3e3) encountered.
TIFFFetchNormalTag: Warning, IO error during reading of "Tag 4"; tag ignored.
TIFFFetchNormalTag: Warning, IO error during reading of "Tag 3"; tag ignored.
TIFFFetchNormalTag: Warning, IO error during reading of "Tag 5"; tag ignored.
TIFFFetchNormalTag: Warning, Incorrect value for "Model"; tag ignored.
TIFFFetchNormalTag: Warning, IO error during reading of "DocumentName"; tag ignored.
TIFFFetchNormalTag: Warning, IO error during reading of "Tag 436"; tag ignored.
TIFFFetchNormalTag: Warning, IO error during reading of "Tag 16384"; tag ignored.
TIFFReadDirectory: Warning, Ignoring ColorMap because BitsPerSample=48>24.
TIFFFetchNormalTag: Warning, Sanity check on size of "Tag 1" value failed; tag ignored.
TIFFFetchNormalTag: Warning, ASCII value for tag "DateTime" contains null byte in value; value incor
TIFFFetchNormalTag: Warning, IO error during reading of "Tag 2"; tag ignored.
TIFFFetchNormalTag: Warning, IO error during reading of "Tag 58339"; tag ignored.
TIFFFetchStripThing: Warning, Incorrect count for "StripOffsets"; tag ignored.
tiffcp: tif_read.c:99: TIFFReadAndRealloc: Assertion `(tif->tif_flags & TIFF_MYBUFFER) != 0' failed.
Aborted
```

Platform

```
$ uname -a
Linux wdw-Precision-Tower-3620 5.13.0-27-generic #29~20.04.1-Ubuntu SMP Fri Jan 14 00:32:30 UTC 2
022 x86_64 x86_64 x86_64 GNU/Linux

# MUST install the libjbig support!
$ sudo apt install -y libjbig-dev
$ CFLAGS="-g -O0" CXXFLAGS="-g -O0" ./configure --disable-shared
$ make -j;make install; make clean
```

poc

Edited 9 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 the assertion failed in tiffcp
!305

tif_jbig.c: fix crash when reading a file with multiple IFD in memory-mapped...
!306

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

Activity

4ugustus @waugustus · 9 months ago

Author Contributor

Hi,

I have analyzed the cause of this crash and found that it is interesting.

Root Cause

The code that triggers the vulnerability is in TIFFReadAndRealloc, tif_read.c:99

```
assert((tif->tif_flags & TIFF_MYBUFFER) != 0);
```

To trigger such vulnerability, we need to answer two questions:

How to make this assertion failed

To make such assertion failed, we need to let (tif->tif_flags & TIFF_MYBUFFER) = 0, which is related to

```
tif->tif_flags &= ~TIFF_MYBUFFER
```

We can find all the places where the tif->tif_flags values have been modified.

```
tif->tif_flags = FILLORDER_MSB2LSB; # tif_open.c:158
tif->tif_flags |= TIFF_MAPPED; # tif_open.c:159
tif->tif_flags |= STRIPCHOP_DEFAULT; # tif_open.c:163
tif->tif_flags |= TIFF_MYBUFFER; # tif_open.c:444
tif->tif_flags |= TIFF_DIRTYDIRECT; # tif_dir.c:748
tif->tif_flags &= ~TIFF_ISTILED; # tif_dir.c:1491
tif->tif_flags |= TIFF_DIRTYDIRECT; # tif_dir.c:748
tif->tif_flags |= TIFF_NOBITREV; # tif_fax3.c:1352
tif->tif_flags &= ~TIFF_DIRTYDIRECT; # tif_dirread.c:4334
tif->tif_flags &= ~TIFF_DIRTYSTRIP; # tif_dirread.c:4335
tif->tif_flags |= TIFF_BUFFERSETUP; # tif_open.c:486
tif->tif_flags &= ~TIFF_MYBUFFER; # tif_read.c:811
tif->tif_flags |= TIFF_BUFFERMMAP; # tif_read.c:823
tif->tif_flags |= TIFF_CODERSETUP; # tif_read.c:1324
tif->tif_flags &= ~(TIFF_NOBITREV|TIFF_NOREADRAW); # tif_compress.c:155
tif->tif_flags |= TIFF_DIRTYDIRECT; # tif_dir.c:748
tif->tif_flags &= ~TIFF_ISTILED; # tif_dir.c:1491
tif->tif_flags |= TIFF_DIRTYDIRECT; # tif_dir.c:748
tif->tif_flags &= ~TIFF_CODERSETUP; # tif_dir.c:239
tif->tif_flags |= TIFF_NOBITREV; # tif_jbig.c:210
tif->tif_flags &= ~TIFF_MAPPED; # tif_jbig.c:211
tif->tif_flags &= ~TIFF_DIRTYDIRECT; # tif_dirread.c:4334
```

```
tif->tif_flags &= ~TIFF_DIRTYSTRIP; # tif_dirread.c:4335
tif->tif_flags &= ~TIFF_BUFFERMMAP; # tif_read.c:850
```

In this PoC, tif_read.c:811 is run. To reach this line, we need to set the condition in tif_read.c:792-794 to True:

```
if (isMapped(tif) &&
    (isFillOrder(tif, td->td_fillorder)
    || (tif->tif_flags & TIFF_NOBITREV)))
```

Which means isMapped(tif) = True, (isFillOrder(tif, td->td_fillorder) || (tif->tif_flags & TIFF_NOBITREV)) = True. From tiffiop.h, we can know

```
#define isMapped(tif) (((tif)->tif_flags & TIFF_MAPPED) != 0)
#define isFillOrder(tif, o) (((tif)->tif_flags & (o)) != 0)
```

And the condition is equal to (tif->tif_flags & TIFF_MAPPED) != 0 and ((tif->tif_flags & td->td_fillorder) != 0 or (tif->tif_flags & TIFF_NOBITREV) != 0). It's easy to know that the first condition is always True. Since td->td_fillorder is user-controllable (Tag 266, 0x010A), we can make the second condition True by controlling the value of td->td_fillorder.

Here, the value of tif->tif_flags is 0x8b11 (1000 1011 0001 0001). We can set the fill order value to 0x1 (0000 0000 0000 0001) to make (tif->tif_flags & td->td_fillorder) True.

```
0A01 0300 0100 0000 0100 0000
```

How to get to this line

Now that we can make the assertion failed, then we need to make the code run on the line where the assertion is located (TIFFReadAndRealloc, tif_read.c:99).

The backtrace of this POC is

```
#0  TIFFReadAndRealloc (tif=0x560368e179f0, size=1, rawdata_offset=0, is_strip=1, strip_or
    module=0x7fa1a9da2af8 <module> "TIFFFillStrip") at tif_read.c:99
#1  0x00007fa1a9d766bc in TIFFReadRawStripOrTile2 (tif=0x560368e179f0, strip_or_tile=0, is
    module=0x7fa1a9da2af8 <module> "TIFFFillStrip") at tif_read.c:669
#2  0x00007fa1a9d76ce4 in TIFFFillStrip (tif=0x560368e179f0, strip=0) at tif_read.c:868
#3  0x00007fa1a9d760ef in TIFFReadEncodedStrip (tif=0x560368e179f0, strip=0, buf=0x560368e
#4  0x0000560366fba0e7 in cpDecodedStrips (in=0x560368e179f0, out=0x560368e17010, imageler
    at tiffcp.c:1155
#5  0x0000560366fb998f in tiffcp (in=0x560368e179f0, out=0x560368e17010) at tiffcp.c:979
#6  0x0000560366fb7fc0 in main (argc=3, argv=0x7ffdc224d7d8) at tiffcp.c:334
```

To reach tif_read.c:99, we need to set the condition in tif_read.c:792-794 to False: (yes, the same condition, but to be false)

```
if (isMapped(tif) &&
    (isFillOrder(tif, td->td_fillorder)
    || (tif->tif_flags & TIFF_NOBITREV)))
```

Besides, we need to set the condition in tif_read.c:853 to False:

```
if( isMapped(tif) )
```

So, we just need to set the condition isMapped(tif) = False to reach the assertion, that is (tif->tif_flags & TIFF_MAPPED) = 0. And it is related to

```
tif->tif_flags &= ~TIFF_MAPPED;
```

which is run in tif_jbig.c:211. To do this, we need to **install the jbig support** and set the compression schema (Tag 259, 0x0103) to JBIG (0x8765)

```
0301 0300 0100 0000 6587 0000
```

Summary

To trigger such vulnerability, we need to

1. set the fillorder field to a number like 0x1
2. set the compression field to 0x8765(JBIG)

How to fix

It is strange that only JBIG's initialization function execute tif->tif_flags &= ~TIFF_MAPPED;

```
/*
 * These flags are set so the JBIG Codec can control when to reverse
 * bits and when not to and to allow the jbig decoder and bit reverser
 * to write to memory when necessary.
 */
tif->tif_flags |= TIFF_NOBITREV;
tif->tif_flags &= ~TIFF_MAPPED;
```

We can remove this line to mitigate the vulnerability.

Edited by 4ugustus 9 months ago

4ugustus mentioned in merge request !305 (closed) 9 months ago

4ugustus changed the description 9 months ago ·

4ugustus changed the description 9 months ago ·

Even Rouault mentioned in merge request !306 (merged) 9 months ago

Even Rouault closed via commit a1c933da 9 months ago

Even Rouault mentioned in commit 5e180045 9 months ago

Even Rouault mentioned in issue #377 (closed) 8 months ago