Heap Buffer Overflow #4729

Closed R0fM1a opened this issue on Jan 19 · 4 comments

R0fM1a commented on Jan 19

ImageMagick version

7.1.0-20

Operating system

Linux

Operating system, version and so on

Linux ubuntu 5.4.0-73-generic #82~18.04.1-Ubuntu SMP Fri Apr 16 15:10:02 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux

Description

Hi, ImageMagick security team
This is ZhangJiaxing (**@R0fM1a**) from Codesafe Team of Legendsec at Qi'anxin Group.
I've found a Heap Buffer Overflow vulnerability in ImageMagick 7.1.0-20.(github commit ID `f54aa4e` in Tue Jan 18 20:00:38 2022 -0500).When someone uses magick to convert a tiff-format image into a picon-format file, the bug will be traggered on.
Please feel free to contact me.
Regards,
ZhangJiaxing

Steps to Reproduce

1. git clone ImageMagick
2. ./configure CC=gcc CFLAGS="-g -fsanitize=address" && make
3. cd utilities &&./magick convert /path/to/poc.tiff output.picon
4. The Asan logs are as follows:
   ==46632==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x62a00000b540 at pc

0x7f7e88ca3257 bp 0x7fffdb9f7370 sp 0x7fffdb9f7360
READ of size 4 at 0x62a00000b540 thread T0
#0 0x7f7e88ca3256 in GetPixelAlpha MagickCore/pixel-accessor.h:59
Added Travis file to allow CI building on Github. #1 0x7f7e88ca763e in WritePICONImage coders/xpm.c:807
Remove files built by build process. #2 0x7f7e885f73ef in WriteImage MagickCore/constitute.c:1221
convert foo.odt foo.pdf fails (delegates do not support shell commands) #3 0x7f7e885f84a0 in WriteImages MagickCore/constitute.c:1442
IM 7 Channel Maps/Masks #4 0x7f7e87e5239f in ConvertImageCommand MagickWand/convert.c:3332
IM7 upgrade notes for changed functions #5 0x7f7e87f604cf in MagickCommandGenesis MagickWand/mogrify.c:188
ImageMagick generates improper output images #6 0x55a7a3ebefcf in MagickMain utilities/magick.c:150
ImageMagick generates improper output images #7 0x55a7a3ebf25a in main utilities/magick.c:182
Remove generated files from IM7 #8 0x7f7e876c2bf6 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21bf6)
Add function getHdriEnabled #9 0x55a7a3ebe9e9 in _start (/home/r0fm1a/ImageMagick/utilities/.libs/magick+0x19e9)

0x62a00000b540 is located 0 bytes to the right of 21312-byte region [0x62a000006200,0x62a00000b540) allocated by thread T0 here:
#0 0x7f7e893e3790 in posix_memalign (/usr/lib/x86_64-linux-gnu/libasan.so.4+0xdf790)
#1 0x7f7e887d1c99 in AcquireAlignedMemory_POSIX MagickCore/memory.c:299
#2 0x7f7e887d1ea8 in AcquireAlignedMemory MagickCore/memory.c:377
#3 0x7f7e88582e0e in OpenPixelCache MagickCore/cache.c:3746
#4 0x7f7e8857b296 in GetImagePixelCache MagickCore/cache.c:1776
#5 0x7f7e8858b2de in SyncImagePixelCache MagickCore/cache.c:5516
#6 0x7f7e88798568 in SetImageStorageClass MagickCore/image.c:2626
#7 0x7f7e885ab718 in AcquireImageColormap MagickCore/colormap.c:152
#8 0x7f7e888731cd in SetGrayscaleImage MagickCore/quantize.c:3772
#9 0x7f7e888714e7 in QuantizeImage MagickCore/quantize.c:3118
#10 0x7f7e88866f5d in CompressImageColormap MagickCore/quantize.c:1204
#11 0x7f7e88ca6f6a in WritePICONImage coders/xpm.c:755
#12 0x7f7e885f73ef in WriteImage MagickCore/constitute.c:1221
#13 0x7f7e885f84a0 in WriteImages MagickCore/constitute.c:1442
#14 0x7f7e87e5239f in ConvertImageCommand MagickWand/convert.c:3332
#15 0x7f7e87f604cf in MagickCommandGenesis MagickWand/mogrify.c:188
#16 0x55a7a3ebefcf in MagickMain utilities/magick.c:150
#17 0x55a7a3ebf25a in main utilities/magick.c:182
#18 0x7f7e876c2bf6 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21bf6)

```
SUMMARY: AddressSanitizer: heap-buffer-overflow MagickCore/pixel-accessor.h:59 in GetPixelAlpha
Shadow bytes around the buggy address:
0x0c547fff9650: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c547fff9660: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c547fff9670: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c547fff9680: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0x0c547fff9690: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c547fff96a0: 00 00 00 00 00 00 00 00[fa]fa fa fa fa fa fa fa
0x0c547fff96b0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c547fff96c0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c547fff96d0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c547fff96e0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c547fff96f0: 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
==46632==ABORTING
```

Images

magick_heap_bof.zip

urban-warrior pushed a commit that referenced this issue on Jan 19

`https://github.com/ImageMagick/ImageMagick/issues/4729` ✓ `e50f19f`

urban-warrior commented on Jan 19 Contributor

Thanks for the problem report. We can reproduce it and will have a patch to fix it in the GIT main branch @ https://github.com/ImageMagick/ImageMagick later today. The patch will be available in the beta releases of ImageMagick @ https://imagemagick.org/download/beta/ by sometime tomorrow.

R0fM1a commented on Jan 23 Author

Do you guys have any trouble reproducing the vulnerability?

urban-warrior commented on Jan 24 Contributor

Why do you ask? In our reply we say "We can reproduce it and will have a patch..."

R0fM1a commented on Jan 24 Author

Emmm... Sorry，I misunderstood your reply as "I will reproduce it and patch...". So embarrassing 😰

R0fM1a closed this as completed on Feb 19

Assignees

No one assigned

Labels

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants