

New issue

Jump to bottom

# AddressSanitizer report LeakSanitizer: detected memory leaks when executing convert command #3540

Closed NISL-SecurityGroup opened this issue on Apr 13, 2021 · 11 comments

NISL-SecurityGro... commented on Apr 13, 2021 • edited

## Prerequisites

I have written a descriptive issue title  
I have searched [open](#) and [closed](#) issues to ensure it has not already been reported  
I have verified that I am using the latest version of ImageMagick

## ImageMagick version

7.0.11-5

## Operating system

Linux

## Operating system, version and so on

Ubuntu 18.04, 64bit

## Description

When we execute the convert command, asan reports the error LeakSanitizer: detected memory leaks.

## Steps to Reproduce

## Command

please run a following cmd with poc file. [POC](#)  
\$ magick convert \$poc out.bmp

## Result

Here's ASAN report:

```
convert: Invalid TIFF directory; tags are not sorted in ascending order. `TIFFReadDirectoryCheckOrder' @ warning/tiff.c/TIFFWarnings/960.
convert: Unknown field with tag 4096 (0x1000) encountered. `TIFFReadDirectory' @ warning/tiff.c/TIFFWarnings/960.
convert: Nonstandard tile width 17990, convert file. `id:000006,sig:09,src:000052,time:16507786,op:flip1,pos:23' @ warning/tiff.c/TIFFWarnings/960.
convert: Unknown field with tag 29255 (0x7247) encountered. `TIFFReadDirectory' @ warning/tiff.c/TIFFWarnings/960.
convert: Unknown field with tag 63745 (0xf901) encountered. `TIFFReadDirectory' @ warning/tiff.c/TIFFWarnings/960.
convert: Unknown field with tag 0 (0x0) encountered. `TIFFReadDirectory' @ warning/tiff.c/TIFFWarnings/960.
convert: ID error during reading of "Tag 4096"; tag ignored. `TIFFFetchNormalTag' @ warning/tiff.c/TIFFWarnings/960.
convert: ASCII value for tag "DocumentName" contains null byte in value; value incorrectly truncated during reading due to implementation limitations. `TIFFFetchNormalTag' @
warning/tiff.c/TIFFWarnings/960.
convert: Incorrect count for "XResolution"; tag ignored. `TIFFFetchNormalTag' @ warning/tiff.c/TIFFWarnings/960.
convert: Incorrect count for "YResolution"; tag ignored. `TIFFFetchNormalTag' @ warning/tiff.c/TIFFWarnings/960.
convert: Sum of Photometric type-related color channels and ExtraSamples doesn't match SamplesPerPixel. Defining non-color channels as ExtraSamples.. `TIFFReadDirectory' @
warning/tiff.c/TIFFWarnings/960.
convert: MaximumChannelsExceeded `id:000006,sig:09,src:000052,time:16507786,op:flip1,pos:23' @ error/tiff.c/ReadTIFFImage/1734.
convert: NoImagesDefined `out.bmp' @ error/convert.c/ConvertImageCommand/3322.
```

```
=====
==9304==ERROR: LeakSanitizer: detected memory leaks
```

```
Direct leak of 152 byte(s) in 1 object(s) allocated from:
#0 0x7fedc6aff517 in malloc (/lib/x86_64-linux-gnu/libasan.so.6+0xb0517)
#1 0x55643dee6ca6 in AcquireMagickMemory MagickCore/memory.c:558
#2 0x55643dee6ccc in AcquireCriticalMemory MagickCore/memory.c:634
#3 0x55643e5a3d32 in AcquireQuantumInfo MagickCore/quantum.c:119
#4 0x55643e309aa6 in ReadTIFFImage coders/tiff.c:1672
#5 0x55643e3c6615 in ReadImage MagickCore/constitute.c:563
#6 0x55643e3c98df in ReadImages MagickCore/constitute.c:955
#7 0x55643e7749c5 in ConvertImageCommand MagickWand/convert.c:611
#8 0x55643e8a5152 in MagickCommandGenesis MagickWand/mogrify.c:191
#9 0x55643deb10fd in MagickMain utilities/magick.c:149
#10 0x55643deb13a8 in main utilities/magick.c:180
#11 0x7fedc57b70b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)
```

```
Indirect leak of 14922012 byte(s) in 28 object(s) allocated from:
#0 0x7fedc6b0021c in posix_memalign (/lib/x86_64-linux-gnu/libasan.so.6+0xb121c)
#1 0x55643dee69c8 in AcquireAlignedMemory_POSIX MagickCore/memory.c:299
#2 0x55643dee6be8 in AcquireAlignedMemory MagickCore/memory.c:377
#3 0x55643dee7348 in AcquireVirtualMemory MagickCore/memory.c:746
#4 0x55643e5a40b4 in AcquireQuantumPixels MagickCore/quantum.c:177
#5 0x55643e5a5db5 in SetQuantumDepth MagickCore/quantum.c:699
#6 0x55643e5a6201 in SetQuantumFormat MagickCore/quantum.c:779
#7 0x55643e309bf7 in ReadTIFFImage coders/tiff.c:1676
#8 0x55643e3c6615 in ReadImage MagickCore/constitute.c:563
#9 0x55643e3c98df in ReadImages MagickCore/constitute.c:955
#10 0x55643e7749c5 in ConvertImageCommand MagickWand/convert.c:611
#11 0x55643e8a5152 in MagickCommandGenesis MagickWand/mogrify.c:191
#12 0x55643deb10fd in MagickMain utilities/magick.c:149
#13 0x55643deb13a8 in main utilities/magick.c:180
#14 0x7fedc57b70b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)
```

```
Indirect leak of 115584 byte(s) in 28 object(s) allocated from:
#0 0x7fedc6b0021c in posix_memalign (/lib/x86_64-linux-gnu/libasan.so.6+0xb121c)
```

```
#1 0x55643dee69c8 in AcquireAlignedMemory_POSIX MagickCore/memory.c:299
#2 0x55643dee6be8 in AcquireAlignedMemory MagickCore/memory.c:377
#3 0x55643dee7137 in AcquireVirtualMemory MagickCore/memory.c:737
#4 0x55643e5a40b4 in AcquireQuantumPixels MagickCore/quantum.c:177
#5 0x55643e5a5db5 in SetQuantumDepth MagickCore/quantum.c:699
#6 0x55643e5a6201 in SetQuantumFormat MagickCore/quantum.c:779
#7 0x55643e309bf7 in ReadTIFFImage coders/tiff.c:1676
#8 0x55643e3c6615 in ReadImage MagickCore/constitute.c:563
#9 0x55643e3c98df in ReadImages MagickCore/constitute.c:955
#10 0x55643e7749c5 in ConvertImageCommand MagickWand/convert.c:611
#11 0x55643e8a5152 in MagickCommandGenesis MagickWand/mogrify.c:191
#12 0x55643deb10fd in MagickMain utilities/magick.c:149
#13 0x55643deb13a8 in main utilities/magick.c:180
#14 0x7fedc57b70b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)

Indirect leak of 224 byte(s) in 1 object(s) allocated from:
#0 0x7fedc6aff517 in malloc (/lib/x86_64-linux-gnu/libasan.so.6+0xb0517)
#1 0x55643dee6ca6 in AcquireMagickMemory MagickCore/memory.c:558
#2 0x55643dee6f29 in AcquireQuantumMemory MagickCore/memory.c:676
#3 0x55643e5a3f68 in AcquireQuantumPixels MagickCore/quantum.c:165
#4 0x55643e5a5db5 in SetQuantumDepth MagickCore/quantum.c:699
#5 0x55643e5a6201 in SetQuantumFormat MagickCore/quantum.c:779
#6 0x55643e309bf7 in ReadTIFFImage coders/tiff.c:1676
#7 0x55643e3c6615 in ReadImage MagickCore/constitute.c:563
#8 0x55643e3c98df in ReadImages MagickCore/constitute.c:955
#9 0x55643e7749c5 in ConvertImageCommand MagickWand/convert.c:611
#10 0x55643e8a5152 in MagickCommandGenesis MagickWand/mogrify.c:191
#11 0x55643deb10fd in MagickMain utilities/magick.c:149
#12 0x55643deb13a8 in main utilities/magick.c:180
#13 0x7fedc57b70b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)

Indirect leak of 64 byte(s) in 1 object(s) allocated from:
#0 0x7fedc6b0021c in posix_memalign (/lib/x86_64-linux-gnu/libasan.so.6+0xb121c)
#1 0x55643df53422 in AcquireSemaphoreMemory MagickCore/semaphore.c:154
#2 0x55643df53523 in AcquireSemaphoreInfo MagickCore/semaphore.c:200
#3 0x55643e5a4cdc in GetQuantumInfo MagickCore/quantum.c:430
#4 0x55643e5a3d82 in AcquireQuantumInfo MagickCore/quantum.c:121
#5 0x55643e309aa6 in ReadTIFFImage coders/tiff.c:1672
#6 0x55643e3c6615 in ReadImage MagickCore/constitute.c:563
#7 0x55643e3c98df in ReadImages MagickCore/constitute.c:955
#8 0x55643e7749c5 in ConvertImageCommand MagickWand/convert.c:611
#9 0x55643e8a5152 in MagickCommandGenesis MagickWand/mogrify.c:191
#10 0x55643deb10fd in MagickMain utilities/magick.c:149
#11 0x55643deb13a8 in main utilities/magick.c:180
#12 0x7fedc57b70b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x270b2)

SUMMARY: AddressSanitizer: 15038036 byte(s) leaked in 59 allocation(s).
```

Additional information:

```
$ CC="gcc" CXX="g++" bash ./configure --disable-shared
$ make

$ export AFL_USE_ASAN=1 AFL_USE_UBSAN=1
$ export AFL_LLVM_CMPLOG=1
$ CC=/usr/local/bin/afl-clang-fast CXX=/usr/local/bin/afl-clang-fast++ bash ./configure --disable-shared
$ make


Version: ImageMagick 7.0.11-5 Q16 x86_64 2021-03-20 https://imagemagick.org
Copyright: (C) 1999-2021 ImageMagick Studio LLC
License: https://imagemagick.org/script/license.php
Features: Cipher DPC HDRI OpenMP(5.0)
Delegates (built-in): bzlib djvu fontconfig freetype jbig jng jpeg lcms lqr lzma openexr png tiff x xml zlib
```

dlemstra commented on Apr 13, 2021 Member

ImageMagick 7.0.11-5 is broken please upgrade to a more recent version and test this again.

 urban-warrior pushed a commit that referenced this issue on Apr 13, 2021 ✓ c6ad94f

<https://github.com/ImageMagick/ImageMagick/issues/3540>

 urban-warrior pushed a commit to ImageMagick/ImageMagick6 that referenced this issue on Apr 13, 2021 ✓ cd7f9fb

<https://github.com/ImageMagick/ImageMagick/issues/3540>

urban-warrior commented on Apr 13, 2021 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.

NISL-SecurityGro... commented on Jun 2, 2021 Author

Can the ImageMagic team assign CVEs to the vulnerabilities we found?


urban-warrior commented on Jun 2, 2021 Contributor

See our [security policy](#).

NISL-SecurityGro... commented on Jun 3, 2021

Author

Thank you. This was assigned [CVE-2021-3574](#).

 NISL-SecurityGroup closed this as completed on Jun 7, 2021

NISL-SecurityGro... commented on Mar 7

Author

It has been verified that the vulnerability has been fixed in the latest version.

rcsanchez97 commented 2 weeks ago

@dlemstra I am trying to figure out how far back the vulnerability affects in terms of older versions. However, since the original proof of concept is no longer accessible (the link returns a 404), I could use some assistance in figuring this out. Looking at the IM6 code, the check that was moved in [cd7f9fb](#) was first introduced in [214d3c3](#). Is it correct to think about this issue as "the call to TIFFClose() needs to happen before trying to work with too many channels?" If so, then I think that the correct way to handle this for versions preceding the introduction of the initial check is to add the check before the `switch (photometric)`?

dlemstra commented 2 weeks ago

Member

I also don't have the the referenced image on my local machine. I am not sure what you are asking but moving this check makes sure that we don't access channels in the image before make sure it's lower/equal to the maximum number of channels that are allowed?

And maybe @NISL-SecurityGroup could add the POC file again?

 rcsanchez97 commented 2 weeks ago

Sorry if I wasn't clear. Essentially, the fix (as shown in [c6ad94f](#) and [cd7f9fb](#)) involves moving this block of code:

```
...
if (samples_per_pixel > MaxPixelChannels)
{
    TIFFClose(tiff);
    ThrowReaderException(CorruptImageError,"MaximumChannelsExceeded");
}
...
```

What I was trying to ask was whether in versions preceding the introduction ([214d3c3](#)) of that particular block of code the memory leak still exists. My instinct is that after tracing through the ASAN report that there would still be memory leak. So, in an older version it would not be so much about moving the aforementioned block of code (which would not be present at all), but rather about introducing it in the correct location.

I agree that if @NISL-SecurityGroup could supply the file again that would be most helpful. That would allow me to test my hypothesis empirically. However, in the event that the file is not provided again, has my attempt at a restatement of my question made it more clear? If it is more clear and you are able to give some guidance, I could proceed based on that and if the POC file is provided again later then I can use that to verify more completely.

dlemstra commented 2 weeks ago

Member

I would prefer it we could wait on the file. I don't have the time to do a deep dive into this without the POC file. That is why I try to add those files to a reported issue when they have an external source but it seems that I forgot to do that here.

 rcsanchez97 commented 2 weeks ago

That's totally understandable. I appreciate your willingness to help. I also have a substantial backlog of issues to review and deal with for the old version I am working with, so waiting on this won't hold up my work in the short term.

Assignees

No one assigned

Labels

None yet

Milestone

No milestone

Development

No branches or pull requests

4 participants

