Project Name

ImageMagick

Source Code Version

7.0.4.1 (7d65a81)
POC Downloadable

CVE ID

CVE-2017-5511

Program Crash Procedure

1. Install Valgrind. (apt-get install valgrind)
2. Download ImageMagick v7.0.4.1
3. Run `./configure`
4. Run `make`
5. Run `make install`

   Compilation Options for ImageMagick (inside Makefile)

   ```
   CC              = gcc -std=gnu99
   CFLAGS          = -fopenmp -g -O2 -Wall -W -pthread
   MAGICK_CFLAGS   = -fopenmp -g -O2 -Wall -W -pthread
   CPPFLAGS        = -I/usr/local/include/ImageMagick
   PCFLAGS         = -fopenmp
   DEFS            = -DHAVE_CONFIG_H
   LDFLAGS         = -lfreetype
   MAGICK_LDFLAGS  = -L/usr/local/lib -lfreetype
   LIBS            = -lMagickCore -llcms -ltiff -lfreetype -ljpeg
                     -lfontconfig -lXext -lSM -lICE -lX11 -lXt -lbz2 -lz
                     -lm -lgomp -lpthread -lltdl
   CXX             = g++
   CXXFLAGS        = -g -O2 -Wall -W -pthread
   ```
6. Run `valgrind magick input.psb null`, where input.psb is attached and null is an empty file

Crash Details

Program Location of Crash

`WritePSDChannel (psd.c:2576)`

Program Location of Root Cause

`ReadPSDLayers (psd.c:1674)`

The crash is caused by a heap buffer overflow vulnerability. Heap overflow is a type of buffer overflow vulnerability, which occurs on the heap area of the memory. When the memory is analyzed by Valgrind, it is shown that the metadata of the heap is corrupted. An attacker may exploit this vulnerability to execute malicious code through the application if no operating system protection is in place, or lead to denial of service conditions.

The magick command attempts to convert files between different image formats and resize them according to its input parameters. In this case, we produce the crash by feeding

ImageMagick a crafted Photoshop (.psb) file. A .psb file contains multiple layers of an image, with its layer names and name length in the header. The input file has a layer with invalid name length.

The crash happened because ImageMagick attempts to read the .psb file's layer names' length. The length is invalid, and it is further used to read and write the layer name. This causes the program to access an invalid memory location during the read/write of the Photoshop file. Further down the line, ImageMagick writes to the output file using the length that was read from the input file. This ultimately triggered the crash of the program.

At first place, the crash occurred because there was an improper cast of the data type. The int type is not casted to unsigned char prior to being assigned.

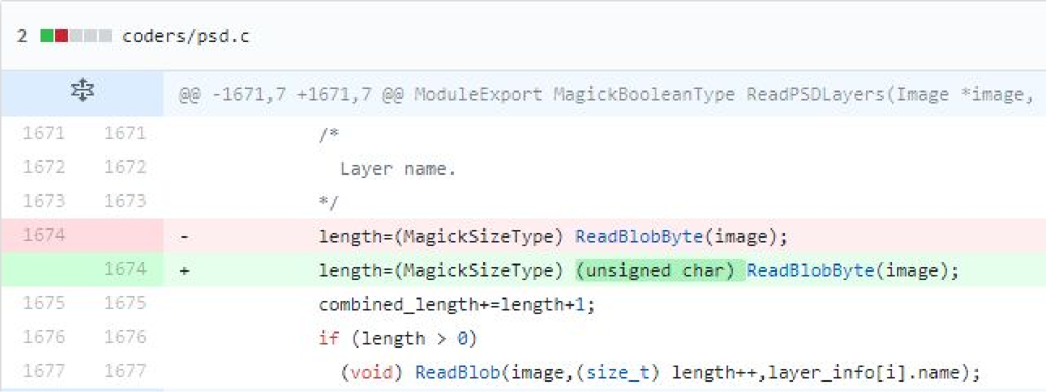
In Valgrind, we get an error of:

```
m_mallocfree.c:303 (get_bszB_as_is): Assertion 'bszB_lo ==
bszB_hi' failed.
```

This error confirms that the program attempts to perform read/write to an invalid position or illegal memory locations.

Bug Fixes

Commit that fixes the bug:
https://github.com/ImageMagick/ImageMagick/commit/7d65a814ac76bd04760072c33e452371692ee790



```
2 coders/psd.c

@@ -1671,7 +1671,7 @@ ModuleExport MagickBooleanType ReadPSDLayers(Image *image,
            /*
              Layer name.
            */
-           length=(MagickSizeType) ReadBlobByte(image);
+           length=(MagickSizeType) (unsigned char) ReadBlobByte(image);
            combined_length+=length+1;
            if (length > 0)
              (void) ReadBlob(image,(size_t) length++,layer_info[i].name);
```

Prior to the fix,there was not any cast to unsigned char before assigning ReadBlobByte(), which is int type, to length. This caused an overflow of the length variable, as explained above. By adding a cast to the unsigned char, the memory is protected from the overflow. The vulnerability affected subsequent operations which relied on the root cause, ultimately leading to an overflow in the heap.

After the fix, an exception is thrown if the length is invalid and the program terminates successfully.

Summary

Improper type casting leading to an overflow is a common vulnerability. Developers need to be careful in changing data types to prevent this problem.

References

https://github.com/ImageMagick/ImageMagick/issues/347
http://www.cvedetails.com/cve-details.php?t=1&cve_id=CVE-2017-5511
https://github.com/ImageMagick/ImageMagick/commit/7d65a814ac76bd04760072c33e452371692ee790