Problem 2

1. Project Name

ImageMagick

2. Source Code Version

Version 7.0.6.0
https://github.com/ImageMagick/ImageMagick/releases/tag/7.0.6-0

3. PoC Downloadable from Internet?

Yes

4. CVE ID

CVE-2017-14684

5. Details Procedures that trigger the crash

a. How the project program is compiled.

The program is compiled using the installation method described by the GitHub source:

1. Go to the directory of the project program and type:
2. make
3. sudo make install
4. magick --version

b. The exact running arguments

1. Add in the input file “poc3.vips”
2. Type: “valgrind convert in/poc3.vips out/poc3”

Crash Description

The crash occurs due to a memory leak in the ResizeMagickMemoryMagickCore. The leak occurs due to uninitialized bytes being used. This exhaust the available system memory as ImageMagick runs therefore causing the program to crash.

The figure below shows where the one of the leaks is coming from.

```
==9264== Syscall param msg->desc.port.name points to uninitialised byte(s)
==9264==    at 0x10084534A: mach_msg_trap (in /usr/lib/system/libsystem_kernel.dylib)
==9264==    by 0x100844796: mach_msg (in /usr/lib/system/libsystem_kernel.dylib)
==9264==    by 0x10083E485: task_set_special_port (in /usr/lib/system/libsystem_kernel.dylib)
==9264==    by 0x1009DA10E: _os_trace_create_debug_control_port (in /usr/lib/system/libsystem_trace.dylib)
==9264==    by 0x1009DA458: _libtrace_init (in /usr/lib/system/libsystem_trace.dylib)
==9264==    by 0x1005439DF: libSystem_initializer (in /usr/lib/libSystem.B.dylib)
==9264==    by 0x100018A1A: ImageLoaderMachO::doModInitFunctions(ImageLoader::LinkContext const&) (in /usr/lib/dyld)
==9264==    by 0x100018C1D: ImageLoaderMachO::doInitialization(ImageLoader::LinkContext const&) (in /usr/lib/dyld)
==9264==    by 0x1000144A9: ImageLoader::recursiveInitialization(ImageLoader::LinkContext const&, unsigned int, char c
==9264==    by 0x100014440: ImageLoader::recursiveInitialization(ImageLoader::LinkContext const&, unsigned int, char c
==9264==    by 0x100013523: ImageLoader::processInitializers(ImageLoader::LinkContext const&, unsigned int, ImageLoade
==9264==    by 0x1000135B8: ImageLoader::runInitializers(ImageLoader::LinkContext const&, ImageLoader::InitializerTim
==9264==  Address 0x10488bd7c is on thread 1's stack
==9264==  in frame #2, created by task_set_special_port (???:)
```

The figure below shows how much memory is being leaked, with 1,834,159 bytes directly lost in 1 block:

```
==9264== HEAP SUMMARY:
==9264==     in use at exit: 1,974,852 bytes in 207 blocks
==9264==   total heap usage: 5,811 allocs, 5,604 frees, 422,130,519 bytes allocated
==9264==
==9264== LEAK SUMMARY:
==9264==    definitely lost: 1,834,159 bytes in 1 blocks
==9264==    indirectly lost: 0 bytes in 0 blocks
==9264==      possibly lost: 72 bytes in 3 blocks
==9264==    still reachable: 122,752 bytes in 49 blocks
==9264==         suppressed: 17,869 bytes in 154 blocks
```

Brief explanation about bug fixed.

The bug is being fixed by checking for the uninitialized variables and destroying them.