Heap-Based Buffer Overflow in TIFFCP.EXE (Neevia docuPrinter Pro)

A vulnerability has been identified in TIFFCP.EXE, a bundled utility within the Neevia docuPrinter Pro suite. This executable statically links to the vulnerable libtiff 3.5.7 library. The vulnerability stems from improper bounds checking during LZW decompression of TIFF files, leading to a heap-based buffer overflow and potential for arbitrary code execution.

Overview

TIFFCP.EXE is included with docuPrinter Pro by Neevia Technology and is used to manipulate TIFF images. Internally, it uses libtiff 3.5.7, a legacy version of the TIFF library with multiple known vulnerabilities. This tool is callable from the command line and is not protected by modern mitigations like ASLR or DEP when built with legacy settings.

Vulnerability Details

When provided with a specially crafted TIFF file using LZW compression, the TIFFCP tool crashes in the LZWDecode function. Specifically, an invalid code offset is used to access the dec_codetab buffer, resulting in a SIGSEGV or STATUS_ACCESS_VIOLATION.

In libtiff/tif_lzw.c:

```
codep = free_entp;
...
t = codep->value; // Crashes here due to invalid codep
```

The application trusts `codep` without validating that it resides within the bounds of `dec_codetab`. This pointer manipulation can lead to arbitrary read/write, making this exploitable under the right conditions.

Reproduction Steps

1. Compile libtiff 3.5.7 with AFL instrumentation on a Linux host:

```
$ CC=afl-clang-fast ./configure
$ make -j$(nproc)
```

2. Fuzz using:

```
$ afl-fuzz -i in -o out -- ./tools/tiffcp @@ /dev/null
```

3. A crash will be generated like:

```
id:000014,sig:11,...
```

4. Debug with:

```
$ gdb --args ./tools/tiffcp crashes/id000014 out.tif
```

Crash Analysis (Linux)

The backtrace in GDB:

```
#6 0x... in LZWDecode (...)
#7 TIFFReadEncodedStrip (...)
#8 cpDecodedStrips (...)
#9 tiffcp (...)
#10 main (...)
```

And the assertion:

```
tif_lzw.c:400: Assertion `&sp->dec_codetab[0] <= free_entp && free_entp <
&sp-
>dec_codetab[CSIZE]' failed.
```

Indicates out-of-bounds access on the heap.

Crash Analysis (Windows)

Running the PoC TIFF file on Windows using:

```
TIFFCP.EXE id000014 out.tif
```

Returns immediately with:

```
> echo %ERRORLEVEL%
-1073741819
```

This is Windows' representation of STATUS_ACCESS_VIOLATION, which aligns with the invalid memory access observed under Linux.

Exploitability

Due to the nature of heap corruption and predictable structure of libtiff internal buffers, this issue may be exploitable to gain arbitrary code execution. At minimum, it results in a denial-of-service via application crash.

Attack Vector

An attacker can send a malicious TIFF file to a user or service that relies on docuPrinter Pro or its TIFFCP.EXE utility. When TIFFCP attempts to process the file (e.g., via conversion pipeline or manual invocation), it will crash and may be exploitable for remote code execution.

Credits

- Discovered by Daniel Conrad
- GitHub: https://github.com/terribledactyl
- LinkedIn: https://www.linkedin.com/in/daniel-conrad-586005258/

References

- https://github.com/terribledactyl/CVE-TIFFCP-OVERFLOW/tree/main
- https://github.com/vadz/libtiff/tree/Release-v3-5-7
- https://neevia.com/products/dppro/