COMMUNITY CONTENT

Home › Submit › 781223

Submit #781223: LibRaw 0.22.0 Out-of-Bounds Read

Title LibRaw 0.22.0 Out-of-Bounds Read

Description Out-of-Bounds Read in LibRaw::nikon_load_padded_packed_raw() Due to Missing Buffer and Dimension Validation

Summary:

A heap out-of-bounds read exists in LibRaw::nikon_load_padded_packed_raw() (src/decoders/decoders_libraw.cpp). The function allocates a row buffer sized from the metadata field load_flags but iterates over image rows using S.raw_width, with no validation that the buffer is large enough for the width-derived access pattern. A crafted TIFF/NEF file with inconsistent load_flags and raw_width metadata triggers the overflow.

Technical Details:

Vulnerability Type: Out-of-Bounds Read (oob_read)
File: decoders/decoders_libraw.cpp
Function: LibRaw::nikon_load_padded_packed_raw()

Vulnerable Code:

```
298: void LibRaw::nikon_load_padded_packed_raw()
299: {
300:   if (libraw_internal_data.unpacker_data.load_flags < 2000 ||
301:       libraw_internal_data.unpacker_data.load_flags > 64000)
302:     return;
303:   unsigned char *buf =
304:       (unsigned char *)calloc(libraw_internal_data.unpacker_data.load_flags, 1);
305:   for (int row = 0; row < S.raw_height; row++)
306:   {
307:     libraw_internal_data.internal_data.input->read(
308:         buf, libraw_internal_data.unpacker_data.load_flags, 1);
309:     for (int icol = 0; icol < S.raw_width / 2; icol++)   // ← loop bound from raw_width
310:     {
311:       imgdata.rawdata.raw_image[(row)*S.raw_width + (icol * 2)] =
312:           ((buf[icol * 3 + 1] & 0xf) << 8) | buf[icol * 3];  // ← OOB read from buf
313:       imgdata.rawdata.raw_image[(row)*S.raw_width + (icol * 2 + 1)] =
314:           buf[icol * 3 + 2] << 4 | ((buf[icol * 3 + 1] & 0xf0) >> 4);
315:     }
316:   }
317:   free(buf);
318: }
```

Community Content

Submissions are made by VulDB community users. VulDB is *not responsible* for their content nor the links to external sources.

Please use the raw information shown and the links listed *with caution*. They might contain malicious and harmful actions, code or data.

The corresponding VulDB entries contain the moderated, verified, and normalized information provided within the raw submission.

Documentation

- Submission Policy
- Data Processing
- CVE Handling

Root Cause:

The buffer is allocated to load_flags bytes. The inner loop runs raw_width / 2 iterations, accessing buf[icol*3], buf[icol*3+1], buf[icol*3+2] — requiring (raw_width/2) * 3 bytes. No check enforces:

load_flags >= (raw_width / 2) * 3

PoC:

A crafted TIFF file: poc_nikonpadded_oob.tif (provided)

Build command used:

```
make -f Makefile.dist \
  CFLAGS="-O1 -I. -w -DUSE_ZLIB -fsanitize=address,undefined -fno-omit-frame-pointer" \
  LDADD="-lz -fsanitize=address,undefined"
```

Note: we use simple_dcraw in this poc but most of the shipped binaries also demonstrate the bug.

./LibRaw/bin/simple_dcraw poc_nikonpadded_oob.tif

Sanitizer Output:

```
=================================================================
==13666==ERROR: AddressSanitizer: heap-buffer-overflow on address 0xe916283e0c60 at pc 0xb52e57765140 bp 0xffffeb439730 sp 0xffffeb439720
READ of size 1 at 0xe916283e0c60 thread T0
    #0 0xb52e5776513c in LibRaw::nikon_load_padded_packed_raw() (/home/roo/Desktop/LibRaw/bin/simple_dcraw+0xf7513c)
    #1 0xb52e57347194 in LibRaw::unpack() (/home/roo/Desktop/LibRaw/bin/simple_dcraw+0xb57194)
    #2 0xb52e57340f58 in main (/home/roo/Desktop/LibRaw/bin/simple_dcraw+0xb50f58)
    #3 0xeb2629242598 in __libc_start_call_main
    #4 0xeb2629242678 in __libc_start_main_impl
    #5 0xb52e5733f56c in _start

0xe916283e0c60 is located 0 bytes after 3040-byte region [0xe916283e0080,0xe916283e0c60)
allocated by thread T0 here:
    #0 0xeb2629f79f44 in calloc
    #1 0xb52e57358b44 in LibRaw::calloc(unsigned long, unsigned long)
    #2 0xb52e57764c80 in LibRaw::nikon_load_padded_packed_raw()
    #3 0xb52e57347194 in LibRaw::unpack()
    #4 0xb52e57340f58 in main

SUMMARY: AddressSanitizer: heap-buffer-overflow in LibRaw::nikon_load_padded_packed_raw()
Shadow bytes around the buggy address:
  0xe916283e0b80: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
=>0xe916283e0c00: 00 00 00 00 00 00 00 00 00 00 00 00[fa]fa fa fa
  0xe916283e0c80: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
==13666==ABORTING
```

Source	⚠ https://github.com/biniamf/pocs/tree/main/libraw_nikonpadded
User	biniam (UID 94731)
Submission	03/16/2026 05:01 PM (22 days ago)
Moderation	04/01/2026 04:52 PM (16 days later)
Status	Accepted
VulDB entry	352671 [LibRaw up to 0.22.0 TIFF/NEF decoders_libraw.cpp nikon_load_padded_packed_raw load_flags/raw_width out-of-bounds]
Points	20