Buffer Over-read in hpjansson/chafa

Valid Reported on Apr 29th 2022

Description

Buffer Over-read in hpjansson/chafa at xwd-loader.c:185

Build

```
export CFLAGS="-g -O0 -lpthread -fsanitize=address"
export CXXFLAGS="-g -O0 -lpthread -fsanitize=address"
export LDFLAGS="-fsanitize=address"

./autogen.sh
./configure --disable-shared

make
```

POC

```
./tools/chafa/chafa ./poc.png
```

poc.png

Asan

```
=================================================================
==599666==ERROR: AddressSanitizer: stack-buffer-overflow on address 0x7fff
READ of size 4 at 0x7fffffffd284 thread T0
    #0 0x4ec1ce in load_header /home/fuzz/fuzz/chafa/tools/c
    #1 0x4eac61 in xwd_loader_new_from_mapping /home/fuzz/f
    #2 0x4e835b in media_loader_new /home/fuzz/fuzz/chafa/tools/chafa/medic
```

```
    #3 0x4d956a in run_generic /home/fuzz/fuzz/chafa/tools/chafa/chafa.c:16
    #4 0x4d8e1c in run /home/fuzz/fuzz/chafa/tools/chafa/chafa.c:1790:12
    #5 0x4cf5ba in run_all /home/fuzz/fuzz/chafa/tools/chafa/chafa.c:1847:2

    #6 0x4cc8ef in main /home/fuzz/fuzz/chafa/tools/chafa/chafa.c:1891:11
    #7 0x7ffff67ab0b2 in __libc_start_main /build/glibc-sMfBJT/glibc-2.31/c
    #8 0x42036d in _start (/home/fuzz/fuzz/chafa/tools/chafa/chafa+0x42036d

Address 0x7fffffffd284 is located in stack of thread T0 at offset 132 in fr
    #0 0x4eb11f in load_header /home/fuzz/fuzz/chafa/tools/chafa/xwd-loader

  This frame has 1 object(s):
    [32, 132) 'in' (line 173) <== Memory access at offset 132 overflows thi
HINT: this may be a false positive if your program uses some custom stack u
      (longjmp and C++ exceptions *are* supported)
SUMMARY: AddressSanitizer: stack-buffer-overflow /home/fuzz/fuzz/chafa/tool
Shadow bytes around the buggy address:
  0x10007fff7a00: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10007fff7a10: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10007fff7a20: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10007fff7a30: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10007fff7a40: f1 f1 f1 f1 00 00 00 00 00 00 00 00 00 00 00 00
=>0x10007fff7a50:[04]f3 f3 f3 f3 f3 f3 00 00 00 00 00 00 00 00
  0x10007fff7a60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10007fff7a70: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10007fff7a80: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10007fff7a90: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x10007fff7aa0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Shadow byte legend (one shadow byte represents 8 application bytes):
  Addressable:           00
  Partially addressable: 01 02 03 04 05 06 07
  Heap left redzone:       fa
  Freed heap region:       fd
  Stack left redzone:      f1
  Stack mid redzone:       f2
  Stack right redzone:     f3
  Stack after return:      f5
  Stack use after scope:   f8
  Global redzone:          f9
  Global init order:       f6
  Poisoned by user:        f7
```

```
  Container overflow:      fc
  Array cookie:            ac
  Intra object redzone:    bb

  ASan internal:           fe
  Left alloca redzone:     ca
  Right alloca redzone:    cb
  Shadow gap:              cc
==599666==ABORTING
```

Impact

This vulnerability is capable of causing a denial of service (crash).

Occurrences

C xwd-loader.c L185

CVE
CVE-2022-2301
(Published)

Vulnerability Type
CWE-126: Buffer Over-read

Severity
Medium (5.5)

Registry
Other

Affected Version
*

Visibility
Public

Status
Fixed

Found by
[redacted] TDHX ICS Security

TDHX ICS Security

@jieyongma

pro

This report was seen 522 times.

We are processing your report and will contact the **hpjansson/chafa** team within 24 hours. 7 months ago

We have contacted a member of the **hpjansson/chafa** team and are waiting to hear back 7 months ago

Hans 7 months ago Maintainer

Good find, thanks. Despite GLib's convention of macro identifiers being uppercase, g_ntohl() is actually a macro. It has multiple implementations, of which one is selected based on the compilation environment: If **OPTIMIZE** is defined and the target is i386 or x86_64, an optimized version is used that evaluates its argument only once. Otherwise a generic implementation is used that evaluates the argument several times, causing the pointer to be incremented repeatedly.

This bug will manifest in unoptimized builds and on non-x86 platforms.

I'll have a fix shortly.

Hans 7 months ago Maintainer

Should be __OPTIMIZE__ above.

We have sent a follow up to the **hpjansson/chafa** team. We will try again in 7 days. 7 months ago

Hans Petter Jansson validated this vulnerability 7 months ago

TDHX ICS Security has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

Hans Petter Jansson marked this as fixed in **1.10.3** with commit **56fabf** 7 months ago

The fix bounty has been dropped ✖

This vulnerability will not receive a CVE ✖

xwd-loader.c#L185 has been validated ✔

TDHX 5 months ago Researcher

@admin can we get a CVE for this? This project is distributed as packages in major linux distros - Debian and such.

Jamie Slome 5 months ago Admin

We sure can - I just require the permission of the maintainer to assign and publish a CVE.

@hpjansson - are you happy for me to assign and publish a CVE for this report?

Hans 5 months ago Maintainer

Be my guest :-)

Jamie Slome 5 months ago Admin

Sorted :)

Sign in to join this conversation

.