huntr

Heap-buffer-overflow in mobi_search_links_kf7 in bfabiszewski/libmobi

0



✓ Valid) Reported on Apr 30th 2022

Description

heap-buffer-overflow /home/ubuntu/libmobi-public/src/parse_rawml.c:110 in mobi_search_links_kf7

Environment

Distributor ID: Ubuntu

Description: Ubuntu 20.04 LTS

Release: 20.04 focal Codename:

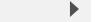
mobitool build: Apr 29 2022 20:52:30 (gcc 9.3.0)

libmobi: 0.10

Build

export CC=gcc CXX=g++ CFLAGS="-fsanitize=address -static-libasan" CXXFLAGS= autogen.sh && ./configure && make





POC

```
./mobitool -e -o ./tmp/ ./poc5
```

poc5

```
==1028892==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x62500 READ of size 1 at 0x62500000748f thread T0
```

```
#0 0x5631f27052e3 in mobi_search_links_kf7 /home/ubuntu/libmobi-public/
#1 0x5631f27052e3 in mobi_search_links_kf7 /home/ubuntu/libmobi-public/
#2 0x5631f2714a31 in mobi_reconstruct_links_kf7 /home/ubuntu/libmobi-pu
#3 0x5631f27180d0 in mobi_reconstruct_links /home/ubuntu/libmobi-public/
#4 0x5631f27180d0 in mobi_parse_rawml_opt /home/ubuntu/libmobi-public/s
#5 0x5631f27180d0 in mobi_parse_rawml /home/ubuntu/libmobi-public/src/p
#6 0x5631f25bbe00 in loadfilename /home/ubuntu/libmobi-public/tools/mobi
#7 0x5631f25bbe00 in main /home/ubuntu/libmobi-public/tools/mobitool.c:
#8 0x7f5bf47900b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6
#9 0x5631f25c76fd in _start (/home/ubuntu/libmobi-public/tools/mobitool
```

0x62500000748f is located 0 bytes to the right of 9103-byte region [0x62500 allocated by thread TO here:

#0 0x5631f26b2748 in malloc (/home/ubuntu/libmobi-public/tools/mobitool #1 0x5631f270cfc0 in mobi_reconstruct_parts /home/ubuntu/libmobi-public

SUMMARY: AddressSanitizer: heap-buffer-overflow /home/ubuntu/libmobi-public Shadow bytes around the buggy address:

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
==1028892==ABORTING
```



Impact

The bug causes the program reads data past the end of the intented buffer. Typically, this can allow attackers to read sensitive information from other memory locations or cause a crash.

Occurrences

c parse_rawml.c L110

CVE

CVE-2022-1908 (Published)

Vulnerability Type

CWE-126: Buffer Over-read

Severity

Low (3.6)

Registry

Other

Affected Version

0.10

Visibility

Status

Found by



cnitlrt master v

Fixed by



Bartek Fabiszewski

unranked V

We are processing your report and will contact the bfabiszewski/libmobi team within 24 hours.

We have contacted a member of the bfabiszewski/libmobi team and are waiting to hear back

Bartek Fabiszewski modified the Severity from High (8.5) to Low (3.6) 7 months ago

Bartek 7 months ago Maintainer

Thanks!

Bartek Fabiszewski validated this vulnerability 7 months ago

cnitIrt has been awarded the disclosure bounty 🗸

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

Bartek Fabiszewski marked this as fixed in 0.11 with commit 1e0378 7 months ago

Bartek Fabiszewski has been awarded the fix bounty 🗸

This vulnerability will not receive a CVE x

parse_rawml.c#L110 has been validated 🗸

Sign in to join this conversation

2022 @ 418sec

huntr

home

hacktivity

leaderboard

FΔO

contact us

terms

privacy policy

part of 418sec

company

about

team