heap-use-after-free in radareorg/radare2

0

Valid Reported on Apr 21st 2022

Description

Whilst experimenting with `radare2`, built from version 5.6.8, we are able to induce a vulnerability at `new_rbtree.c:411` in function `r_rbnode_next`, using `radare2` as a harness.

```
409:  R_API RRBNode *r_rbnode_next(RRBNode *node) {
410:    r_return_val_if_fail (node, NULL);
//use-after-free here
411:    if (node->link[1]) {
412:        node = node->link[1];
413:        while (node->link[0]) {
414:            node = node->link[0];
415:        }
416:        return node;
417:    }
418:    RRBNode *parent = node->parent;
419:    while (parent && parent->link[1] == node) {
420:        node = parent;
421:        parent = node->parent;
422:    }
423:    return parent;
424:  }
```

Due to not properly handling pointers, a heap-based use-after-free will be triggered when the software encounters a malformed file, which could result in denial of service.
We found that the vulnerability exists in the latest master branch as well.

Environment

Ubuntu 20.04 LTS x86_64
gcc 10.3.0

Proof of Concept

The POC is: poc
The reproducing process is:

```
# build with address sanitizer
SANITIZE=address ./sys/sanitize.sh
# disable some features of address sanitizer to avoid false positives
export ASAN_OPTIONS=detect_leaks=0:abort_on_error=1:symbolize=1:allocator_m
# trigger the crash
./radare2 -A -q POC_FILE
```

The ASAN report is:

```
=================================================================
==119195==ERROR: AddressSanitizer: heap-use-after-free on address 0x604001a
READ of size 8 at 0x604001aab958 thread T0
    #0 0x7ffff73a7e3c in r_rbnode_next /work/libraries/radare2-version/libr
    #1 0x7ffff6c489a6 in r_io_bank_map_add_top /work/libraries/radare2-vers
    #2 0x7ffff6c37454 in r_io_map_add /work/libraries/radare2-version/libr/
    #3 0x7ffff46dc4ba in add_section /work/libraries/radare2-version/libr/c
    #4 0x7ffff46e0134 in bin_sections /work/libraries/radare2-version/libr/
    #5 0x7ffff46e7f24 in r_core_bin_info /work/libraries/radare2-version/li
    #6 0x7ffff46c9a6b in r_core_bin_set_env /work/libraries/radare2-version
    #7 0x7ffff463e893 in r_core_file_do_load_for_io_plugin /work/libraries/
    #8 0x7ffff464014f in r_core_bin_load /work/libraries/radare2-version/li
    #9 0x7ffff7182a0d in r_main_radare2 /work/libraries/radare2-version/lib
    #10 0x5555555556ff in main /work/libraries/radare2-version/binr/radare2
    #11 0x7ffff6f6a0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.
    #12 0x55555555528d in _start (/work/libraries/radare2-version/binr/rada

0x604001aab958 is located 8 bytes inside of 40-byte region [0x604001aab950,
freed by thread T0 here:
    #0 0x7ffff769b8f7 in __interceptor_free ../../../../src/libsanitizer/as
    #1 0x7ffff73a786f in r_crbtree_take /work/libraries/rad
    #2 0x7ffff73a7b53 in r_crbtree_delete /work/libraries/r
    #3 0x7ffff6c48a5d in r_io_bank_map_add_top /work/libraries/radare2-vers
```

```
    #4 0x7ffff6c37454 in r_io_map_add /work/libraries/radare2-version/libr/
    #5 0x7ffff46dc4ba in add_section /work/libraries/radare2-version/libr/c
    #6 0x7ffff46e0134 in bin_sections /work/libraries/radare2-version/libr/

    #7 0x7ffff46e7f24 in r_core_bin_info /work/libraries/radare2-version/li
    #8 0x7ffff46c9a6b in r_core_bin_set_env /work/libraries/radare2-version
    #9 0x7ffff463e893 in r_core_file_do_load_for_io_plugin /work/libraries/
    #10 0x7ffff464014f in r_core_bin_load /work/libraries/radare2-version/l
    #11 0x7ffff7182a0d in r_main_radare2 /work/libraries/radare2-version/li
    #12 0x5555555556ff in main /work/libraries/radare2-version/binr/radare2
    #13 0x7ffff6f6a0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.

previously allocated by thread T0 here:
    #0 0x7ffff769be17 in __interceptor_calloc ../../../../src/libsanitizer/
    #1 0x7ffff73a5308 in _node_new /work/libraries/radare2-version/libr/uti
    #2 0x7ffff73a58cd in r_crbtree_insert /work/libraries/radare2-version/l
    #3 0x7ffff6c48307 in r_io_bank_map_add_top /work/libraries/radare2-vers
    #4 0x7ffff6c37454 in r_io_map_add /work/libraries/radare2-version/libr/
    #5 0x7ffff6c30f3d in r_io_open_at /work/libraries/radare2-version/libr/
    #6 0x7ffff46dc047 in io_create_mem_map /work/libraries/radare2-version/
    #7 0x7ffff46dc318 in add_section /work/libraries/radare2-version/libr/c
    #8 0x7ffff46e0134 in bin_sections /work/libraries/radare2-version/libr/
    #9 0x7ffff46e7f24 in r_core_bin_info /work/libraries/radare2-version/li
    #10 0x7ffff46c9a6b in r_core_bin_set_env /work/libraries/radare2-versio
    #11 0x7ffff463e893 in r_core_file_do_load_for_io_plugin /work/libraries
    #12 0x7ffff464014f in r_core_bin_load /work/libraries/radare2-version/l
    #13 0x7ffff7182a0d in r_main_radare2 /work/libraries/radare2-version/li
    #14 0x5555555556ff in main /work/libraries/radare2-version/binr/radare2
    #15 0x7ffff6f6a0b2 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.

SUMMARY: AddressSanitizer: heap-use-after-free /work/libraries/radare2-vers
Shadow bytes around the buggy address:
  0x0c088034d6d0: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
  0x0c088034d6e0: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
  0x0c088034d6f0: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
  0x0c088034d700: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
  0x0c088034d710: fa fa fd fd fd fd fd fa fa fa 00 00 00 00 00 fa
=>0x0c088034d720: fa fa 00 00 00 00 00 fa fa fa fd[fd]fd fd fd fa
  0x0c088034d730: fa fa fd fd fd fd fd fa fa fa 00 00 00 00 00 fa
  0x0c088034d740: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
  0x0c088034d750: fa fa 00 00 00 00 00 fa fa fa 00 00 00 00 00 fa
```

```
  0x0c088034d760: fa fa 00 00 00 00 00 fa fa fa 00 00 00 00 00 fa
  0x0c088034d770: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
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
==119195==ABORTING
Aborted
```

Acknowledgements

This vulnerability was found by Xingyuan Mo from 360 IceSword Lab

Impact

This vulnerability is capable of inducing denial of service.

CVE
CVE-2022-1444
(Published)

Vulnerability Type
CWE-416: Use After Free

Severity
High (7.5)

Registry
Other

Affected Version
5.6.8

Visibility
Public

Status
Fixed

Found by

hdthky
@hdthky
unranked

This report was seen 559 times.

We are processing your report and will contact the **radareorg/radare2** team within 24 hours. 7 months ago

hdthky modified the report 7 months ago

hdthky modified the report 7 months ago

We have contacted a member of the **radareorg/radare2** team and are waiting to hear back 7 months ago

pancake validated this vulnerability 7 months ago

hdthky has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

The researcher's credibility has increased: +7

pancake marked this as fixed in **5.7.0** with commit **141897** 7 months ago

The fix bounty has been dropped ✖

This vulnerability will not receive a CVE ✖

pancake 7 months ago Maintainer

The fix was made by condret, not sure how can i give him perms to request the bounties

Sign in to join this conversation