axiomatic-systems / **Bento4** Public

Code Issues 421 Pull requests 27 Actions Projects Wiki ...

New issue

Jump to bottom

A heap-buffer-overflow in Avcinfo #780

Open **DylanSec** opened this issue on Sep 27 · 3 comments

DylanSec commented on Sep 27 • edited

Summary

Hi, developers of Bento4:
I tested the binary Avcinfo with my fuzzer, and a crash incurred—heap-buffer-overflow. The following is the details. I think this error is different from both #731 and #610.

Bug

Detected heap-buffer-overflow in Avcinfo.

```
root@4w41awdas71:/# ./Bento4/cmakebuild/avcinfo fuzz-avcinfo/out/crashes/POC_avcinfo_15644345
=================================================================
==708228==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x602000000011 at pc
0x0000004fb133 bp 0x7ffea9099cb0 sp 0x7ffea9099ca8
READ of size 1 at 0x602000000011 thread T0
    #0 0x4fb132 in main (/Bento4/cmakebuild/avcinfo+0x4fb132)
    #1 0x7fa90b673c86 in __libc_start_main (/lib/x86_64-linux-gnu/libc.so.6+0x21c86)
    #2 0x41d5a9 in _start (/Bento4/cmakebuild/avcinfo+0x41d5a9)

0x602000000011 is located 0 bytes to the right of 1-byte region [0x602000000010,0x602000000011)
allocated by thread T0 here:
    #0 0x4f58a8 in operator new[](unsigned long) /llvm-project/compiler-
rt/lib/asan/asan_new_delete.cpp:102
    #1 0x4fb503 in AP4_DataBuffer::SetDataSize(unsigned int) (/Bento4/cmakebuild/avcinfo+0x4fb503)

SUMMARY: AddressSanitizer: heap-buffer-overflow (/Bento4/cmakebuild/avcinfo+0x4fb132) in main
Shadow bytes around the buggy address:
  0x0c047fff7fb0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7fc0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7fd0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7fe0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0c047fff7ff0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

```
=>0x0c047fff8000: fa fa[01]fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8010: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8020: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8030: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8040: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0c047fff8050: 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
==708228==ABORTING
```

POC

POC_avcinfo_15644345.zip

Environment

Ubuntu 18.04.6 LTS (docker)
clang 12.0.1
clang++ 12.0.1
Bento4 master branch(`5b7cc25`) && Bento4 release version(1.6.0-639)

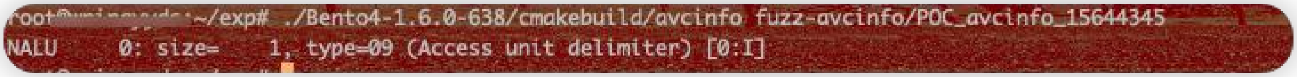
Credit

Xudong Cao (NCNIPC of China)
Jiayuan Zhang (NCNIPC of China)
Han Zheng (NCNIPC of China, Hexhive)

Thank you for your time!

DylanSec changed the title ~~Heap-buffer-overflow~~ **A heap-buffer-overflow in Avcinfo** on Sep 27

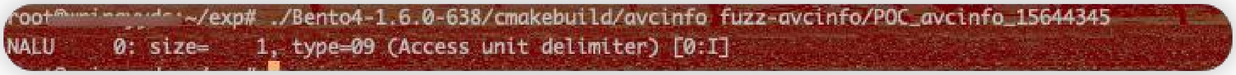
Wninayyds commented on Oct 10



```
root@[redacted]~/exp# ./Bento4-1.6.0-638/cmakebuild/avcinfo fuzz-avcinfo/POC_avcinfo_15644345
NALU     0: size=    1, type=09 (Access unit delimiter) [0:I]
```

why?

DylanSec commented on Oct 11 Author

> 
>
> ```
> root@[redacted]~/exp# ./Bento4-1.6.0-638/cmakebuild/avcinfo fuzz-avcinfo/POC_avcinfo_15644345
> NALU     0: size=    1, type=09 (Access unit delimiter) [0:I]
> ```
>
> why?

Thanks for your comment, I didn't try to trigger this crash in version 638, maybe you can try 639 or the latest master branch.

Wninayyds commented on Oct 11

thx!

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants