There are some vulnerabilities in Bento4 #779

Open **yuhanghuang** opened this issue on Sep 27 · 0 comments

yuhanghuang commented on Sep 27 • edited

Hello, I use my fuzzer to fuzz binary mp4tag and binary mp42hevc , and found some crashes. The bug1 is different from issue #295, because i run the test-001.mp4 finding it useless. Here are the details.

Bug1

```
┌──(kali㉿kali)-[~/Desktop/Bento4/cmakebuild]
└─$ ./mp4tag mp4tag_poc
ERROR: cannot open input file

=================================================================
==2376684==ERROR: LeakSanitizer: detected memory leaks

Direct leak of 40 byte(s) in 1 object(s) allocated from:
    #0 0x4c93dd in operator new(unsigned long)
(/home/kali/Desktop/Bento4/cmakebuild/mp4tag+0x4c93dd)
    #1 0x4ccf5e in ParseCommandLine(int, char**)
/home/kali/Desktop/Bento4/Source/C++/Apps/Mp4Tag/Mp4Tag.cpp:207:34
    #2 0x4ccf5e in main /home/kali/Desktop/Bento4/Source/C++/Apps/Mp4Tag/Mp4Tag.cpp:783:5
    #3 0x7f1b3ea14209 in __libc_start_call_main csu/../sysdeps/nptl/libc_start_call_main.h:58:16

SUMMARY: AddressSanitizer: 40 byte(s) leaked in 1 allocation(s).
```

Bug2

```
┌──(kali㉿kali)-[~/Desktop/Bento4/cmakebuild]
└─$ ./mp42hevc mp42hevc_poc /dev/null
1 ×
```

```
ERROR: cannot open input (-5)
AddressSanitizer:DEADLYSIGNAL
=================================================================
==2392528==ERROR: AddressSanitizer: SEGV on unknown address 0x000000000000 (pc 0x0000004d52c3 bp
0x7fff8ac3ad90 sp 0x7fff8ac3ac40 T0)
==2392528==The signal is caused by a READ memory access.
==2392528==Hint: address points to the zero page.
    #0 0x4d52c3 in AP4_File::ParseStream(AP4_ByteStream&, AP4_AtomFactory&, bool)
/home/kali/Desktop/Bento4/Source/C++/Core/Ap4File.cpp:103:12
    #1 0x4d5aea in AP4_File::AP4_File(AP4_ByteStream&, bool)
/home/kali/Desktop/Bento4/Source/C++/Core/Ap4File.cpp:78:5
    #2 0x4cbea4 in main /home/kali/Desktop/Bento4/Source/C++/Apps/Mp42Hevc/Mp42Hevc.cpp:374:32
    #3 0x7fd8587a8209 in __libc_start_call_main csu/../sysdeps/nptl/libc_start_call_main.h:58:16
    #4 0x7fd8587a82bb in __libc_start_main csu/../csu/libc-start.c:389:3
    #5 0x41f600 in _start (/home/kali/Desktop/Bento4/cmakebuild/mp42hevc+0x41f600)

AddressSanitizer can not provide additional info.
SUMMARY: AddressSanitizer: SEGV /home/kali/Desktop/Bento4/Source/C++/Core/Ap4File.cpp:103:12 in
AP4_File::ParseStream(AP4_ByteStream&, AP4_AtomFactory&, bool)
==2392528==ABORTING
```

Environment

```
clang 11.0.1
clang++ 11.0.1
version:master branch(commit[5b7cc25](https://github.com/axiomatic-
systems/Bento4/commit/5b7cc2500d514717a64675fcf631939494c074ce))+Bento4-1.6.0-639
```

Platform

```
└─$ uname -a
1 ×
Linux kali 5.10.0-kali9-amd64 #1 SMP Debian 5.10.46-4kali1 (2021-08-09) x86_64 GNU/Linux
```

How to reproduce

```
export CC=clang
export CXX=clang++
```

```
export CFLAGS="-fsanitize=address -g"
export CXXFLAGS="-fsanitize=address -g"
mkdir cmakebuild
cd cmakebuild
cmake -DCMAKE_BUILD_TYPE=Release ..
make
```

Note

```
I find the two bugs not only exist in latest branch but also exist in latest release version
Bento4-1.6.0-639.
```

POC

poc_Bento4.zip

Credit

```
Yuhang Huang ([NCNIPC of China](http://www.nipc.org.cn/))
Han Zheng ([NCNIPC of China](http://www.nipc.org.cn/), [Hexhive](http://hexhive.epfl.ch/))
Wanying Cao, Mengyue Feng([NCNIPC of China](http://www.nipc.org.cn/))

Thansk for your time!
```

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant