New issue

Jump to bottom

A global-buffer-overflow in Ap4ByteStream.cpp:783:5 #545

Closed seviezhou opened this issue on Aug 21, 2020 · 1 comment

seviezhou commented on Aug 21, 2020

System info

Ubuntu x86_64, clang 6.0, mp42aac (latest master 174b94)

Configure

cmake .. -DCMAKE_CXX_FLAGS="-fsanitize=address -g" -DCMAKE_C_FLAGS="-fsanitize=address -g" -DCMAKE_EXE_LINKER_FLAGS="-fsanitize=address" -DCMAKE_MODULE_LINKER_FLAGS="-fsanitize=address"

Command line

./build/mp4info --show-layout --show-samples --show-sample-data @@

AddressSanitizer output

```
=================================================================
==47025==ERROR: AddressSanitizer: global-buffer-overflow on address 0x0000016a3561 at pc 0x0000004d9dd2 bp 0x7ffec88c7210 sp 0x7ffec88c69c0
READ of size 243 at 0x0000016a3561 thread T0
    #0 0x4d9dd1 in __asan_memcpy (/home/seviezhou/bento4/build/mp4info+0x4d9dd1)
    #1 0x56de75 in AP4_MemoryByteStream::WritePartial(void const*, unsigned int, unsigned int&) /home/seviezhou/Bento4/Source/C++/Core/Ap4ByteStream.cpp:783:5
    #2 0x5681db in AP4_ByteStream::Write(void const*, unsigned int) /home/seviezhou/Bento4/Source/C++/Core/Ap4ByteStream.cpp:77:29
    #3 0x58f7f3 in AP4_HdlrAtom::WriteFields(AP4_ByteStream&) /home/seviezhou/Bento4/Source/C++/Core/Ap4HdlrAtom.cpp:141:29
    #4 0x54a666 in AP4_Atom::Write(AP4_ByteStream&) /home/seviezhou/Bento4/Source/C++/Core/Ap4Atom.cpp:229:14
    #5 0x54b09a in AP4_Atom::Clone() /home/seviezhou/Bento4/Source/C++/Core/Ap4Atom.cpp:316:9
    #6 0x60a91b in AP4_SampleDescription::AP4_SampleDescription(AP4_SampleDescription::Type, unsigned int, AP4_AtomParent*)
/home/seviezhou/Bento4/Source/C++/Core/Ap4SampleDescription.cpp:132:41
    #7 0x61a658 in AP4_GenericAudioSampleDescription::AP4_GenericAudioSampleDescription(unsigned int, unsigned int, unsigned short, unsigned short, AP4_AtomParent*)
/home/seviezhou/Bento4/Source/C++/Core/Ap4SampleDescription.h:248:9
    #8 0x61a658 in AP4_AudioSampleEntry::ToSampleDescription() /home/seviezhou/Bento4/Source/C++/Core/Ap4SampleEntry.cpp:625
    #9 0x63a26e in AP4_StsdAtom::GetSampleDescription(unsigned int) /home/seviezhou/Bento4/Source/C++/Core/Ap4StsdAtom.cpp:181:53
    #10 0x69f05e in AP4_AtomSampleTable::GetSampleDescription(unsigned int) /home/seviezhou/Bento4/Source/C++/Core/Ap4AtomSampleTable.cpp:207:37
    #11 0x65a9cd in AP4_Track::GetSampleDescription(unsigned int) /home/seviezhou/Bento4/Source/C++/Core/Ap4Track.cpp:445:43
    #12 0x51e2f4 in ShowTrackInfo_Text(AP4_Movie&, AP4_Track&, AP4_ByteStream&, bool, bool, bool, bool) /home/seviezhou/Bento4/Source/C++/Apps/Mp4Info/Mp4Info.cpp:1239:52
    #13 0x51e2f4 in ShowTrackInfo(AP4_Movie&, AP4_Track&, AP4_ByteStream&, bool, bool, bool, bool) /home/seviezhou/Bento4/Source/C++/Apps/Mp4Info/Mp4Info.cpp:1363
    #14 0x51d596 in ShowTracks(AP4_Movie&, AP4_List<AP4_Track>&, AP4_ByteStream&, bool, bool, bool, bool) /home/seviezhou/Bento4/Source/C++/Apps/Mp4Info/Mp4Info.cpp:1473:9
    #15 0x519316 in main /home/seviezhou/Bento4/Source/C++/Apps/Mp4Info/Mp4Info.cpp:1755:13
    #16 0x7eff145b3b96 in __libc_start_main /build/glibc-OTsEL5/glibc-2.27/csu/../csu/libc-start.c:310
    #17 0x41b059 in _start (/home/seviezhou/bento4/build/mp4info+0x41b059)

0x0000016a3561 is located 63 bytes to the left of global variable 'AP4_GlobalOptions::g_Entries' defined in '/home/seviezhou/Bento4/Source/C++/Core/Ap4Utils.cpp:37:56' (0x16a35a0)
of size 8
0x0000016a3561 is located 0 bytes to the right of global variable 'AP4_String::EmptyString' defined in '/home/seviezhou/Bento4/Source/C++/Core/Ap4String.cpp:39:18' (0x16a3560) of
size 1
  'AP4_String::EmptyString' is ascii string ''
SUMMARY: AddressSanitizer: global-buffer-overflow (/home/seviezhou/bento4/build/mp4info+0x4d9dd1) in __asan_memcpy
Shadow bytes around the buggy address:
  0x0000802cc650: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000802cc660: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000802cc670: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 04 f9
  0x0000802cc680: f9 f9 f9 f9 00 00 00 00 00 00 00 00 00 00 f9 f9
  0x0000802cc690: f9 f9 f9 f9 00 00 00 00 00 00 00 00 00 00 f9 f9
=>0x0000802cc6a0: f9 f9 f9 f9 00 00 f9 f9 f9 f9 f9 f9[01]f9 f9 f9
  0x0000802cc6b0: f9 f9 f9 f9 00 f9 f9 f9 f9 f9 f9 f9 00 00 00 f9
  0x0000802cc6c0: f9 f9 f9 f9 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000802cc6d0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000802cc6e0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000802cc6f0: 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
==47025==ABORTING
```

POC

global-overflow-WritePartial-Ap4ByteStream-783.zip

X3eRo0 mentioned this issue on Jul 22, 2021

Global Buffer Overflow in mp4info #626

Closed

biscrblx commented on Mar 21

This issue was brought to our attention by https://nvd.nist.gov/vuln/detail/CVE-2021-32265.

In a situation where a hdlr atom has no name defined, it is possible for this overflow to be generated.

Analysis of the issue

The source stream contains multiple 'hdlr' atoms. All of handler_type 'vide'. One of these has a name, but the others don't. If no name is attached to the hdlr atom, then this access violation will occur due to the code in AP4_HdlrAtom::WriteFields, approximately line 138.

The failure is if name_size==0 (as retrieved at line 123 of this method) AND the size of the source atom is short enough that it shouldn't contain a name (which is valid); then it will recalculate name_size at line 138; giving it a negative value - but due to it being a u8, it ends up being 237 characters in length, instead of 0.

Resolution

Check for name_size before writing anything; if it's zero, then do not write any further data to this atom.

Diff of a fix attached.

CVE-2021-32265.txt

biscrblx added a commit to biscrblx/Bento4 that referenced this issue on Mar 21

`If there is no name, do not write any further data to the atom.` ··· `447bbfd`

biscrblx mentioned this issue on Mar 21

A global-buffer-overflow in Ap4ByteStream.cpp:783 as reported on issue 545. #685

Closed

barbibulle closed this as completed in `1954c9e` on May 1

CastagnaIT pushed a commit to CastagnaIT/Bento4 that referenced this issue on Jul 3

`If there is no name, do not write any further data to the atom. Fix for` ··· `ab282fe`

CastagnaIT pushed a commit to xbmc/Bento4 that referenced this issue on Jul 3

`fix axiomatic-systems#545` `ebc76a9`

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants