New issue

Jump to bottom

# Heap-buffer-overflow in /fuzz_software/tsMuxer-master/tsMuxer/bitStream.h:224:37 in BitStreamWriter::flushBits() and BitStreamReader::getCurVal #641

Closed **yangfar** opened this issue on Oct 15 · 5 comments

Labels bug

**yangfar** commented on Oct 15 • edited

## Version

tsMuxeR version 2.6.16-dev. github.com/justdan96/tsMuxer
https://github.com/justdan96/tsMuxer/commit/fc36229b007d476437271e03e5297b0f04f61ed6

## Description

## Crash1

```
=================================================================
==1445939==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x60400001d9b1 at pc
0x000000652ad9 bp 0x7ffe58ccc330 sp 0x7ffe58ccc328
READ of size 4 at 0x60400001d9b1 thread T0
#0 0x652ad8 in BitStreamWriter::flushBits() /home/hjsz/fuzz_software/tsMuxer-
master/tsMuxer/bitStream.h:224:37
#1 0x652ad8 in HevcUnit::updateBits(int, int, int) /home/hjsz/fuzz_software/tsMuxer-
master/tsMuxer/hevc.cpp:89:15
#2 0x66a7b9 in HEVCStreamReader::updateStreamFps(void*, unsigned char*, unsigned char*, int)
/home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/hevcStreamReader.cpp:372:10
#3 0x7c8bd0 in MPEGStreamReader::updateFPS(void*, unsigned char*, unsigned char*, int)
/home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/mpegStreamReader.cpp:310:9
#4 0x665b53 in HEVCStreamReader::checkStream(unsigned char*, int) /home/hjsz/fuzz_software/tsMuxer-
master/tsMuxer/hevcStreamReader.cpp:68:17
#5 0x7386cd in METADemuxer::detectTrackReader(unsigned char*, int, AbstractStreamReader::ContainerType,
int, int) /home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/metaDemuxer.cpp:796:22
#6 0x7308c4 in METADemuxer::DetectStreamReader(BufferedReaderManager&,
std::__cxx11::basic_string<char, std::char_traits, std::allocator > const&, bool)
/home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/metaDemuxer.cpp:696:35
#7 0x6a9a25 in detectStreamReader(char const*, MPLSParser*, bool) /home/hjsz/fuzz_software/tsMuxer-
master/tsMuxer/main.cpp:120:34
#8 0x6b6504 in main /home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/main.cpp:700:17
#9 0x7fa8fce89082 in __libc_start_main /build/glibc-SzIz7B/glibc-2.31/csu/../csu/libc-start.c:308:16
#10 0x50804d in _start (/home/hjsz/fuzz_software/tsMuxer-master/build/tsMuxer/tsmuxer+0x50804d)

0x60400001d9b3 is located 0 bytes to the right of 35-byte region [0x60400001d990,0x60400001d9b3)
allocated by thread T0 here:
#0 0x5b000d in operator new[](unsigned long) (/home/hjsz/fuzz_software/tsMuxer-
master/build/tsMuxer/tsmuxer+0x5b000d)
#1 0x651f6b in HevcUnit::decodeBuffer(unsigned char const*, unsigned char const*)
/home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/hevc.cpp:40:19
```

SUMMARY: AddressSanitizer: heap-buffer-overflow /home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/bitStream.h:224:37 in BitStreamWriter::flushBits()
Shadow bytes around the buggy address:
0x0c087fffbae0: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
0x0c087fffbaf0: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
0x0c087fffbb00: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
0x0c087fffbb10: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
0x0c087fffbb20: fa fa fd fd fd fd fd fa fa fa fd fd fd fd fd fa
=>0x0c087fffbb30: fa fa 00 00 00 00[03]fa fa fa fa fa fa fa fa
0x0c087fffbb40: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c087fffbb50: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c087fffbb60: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c087fffbb70: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c087fffbb80: 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
==1445939==ABORTING

## Crash2

**tsMuxeR version 2.6.16-dev. github.com/justdan96/tsMuxer**

```
=================================================================
==1450879==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x608000004b7e at pc
0x0000005b633f bp 0x7fff15480e20 sp 0x7fff15480e18
READ of size 1 at 0x608000004b7e thread T0
#0 0x5b633e in BitStreamReader::getCurVal(unsigned int*) /home/hjsz/fuzz_software/tsMuxer-
master/tsMuxer/bitStream.h:60:20
#1 0x7f9bbe in BitStreamReader::setBuffer(unsigned char*, unsigned char*)
/home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/bitStream.h:70:20
#2 0x7f9bbe in SEIUnit::pic_timing(SPSUnit&, unsigned char*, int, bool) /home/hjsz/fuzz_software/tsMuxer-
master/tsMuxer/nalUnits.cpp:1693:15
#3 0x7f8166 in SEIUnit::sei_payload(SPSUnit&, int, unsigned char*, int, int)
/home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/nalUnits.cpp:1531:9
#4 0x7f74af in SEIUnit::deserialize(SPSUnit&, int) /home/hjsz/fuzz_software/tsMuxer-
master/tsMuxer/nalUnits.cpp:1408:13
#5 0x62acfb in H264StreamReader::checkStream(unsigned char*, int) /home/hjsz/fuzz_software/tsMuxer-
master/tsMuxer/h264StreamReader.cpp:138:25
#6 0x737282 in METADemuxer::detectTrackReader(unsigned char*, int, AbstractStreamReader::ContainerType,
int, int) /home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/metaDemuxer.cpp:760:22
#7 0x7308c4 in METADemuxer::DetectStreamReader(BufferedReaderManager&,
std::__cxx11::basic_string<char, std::char_traits, std::allocator > const&, bool)
/home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/metaDemuxer.cpp:696:35
#8 0x6a9a25 in detectStreamReader(char const*, MPLSParser*, bool) /home/hjsz/fuzz_software/tsMuxer-
master/tsMuxer/main.cpp:120:34
#9 0x6b6504 in main /home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/main.cpp:700:17
#10 0x7f20cbc80082 in __libc_start_main /build/glibc-SzIz7B/glibc-2.31/csu/../csu/libc-start.c:308:16
#11 0x50804d in _start (/home/hjsz/fuzz_software/tsMuxer-master/build/tsMuxer/tsmuxer+0x50804d)

0x608000004b7e is located 0 bytes to the right of 94-byte region [0x608000004b20,0x608000004b7e)
allocated by thread T0 here:
#0 0x5b000d in operator new[](unsigned long) (/home/hjsz/fuzz_software/tsMuxer-
master/build/tsMuxer/tsmuxer+0x5b000d)
#1 0x7e264b in NALUnit::decodeBuffer(unsigned char const*, unsigned char const*)
/home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/nalUnits.cpp:271:19
```

```
SUMMARY: AddressSanitizer: heap-buffer-overflow /home/hjsz/fuzz_software/tsMuxer-master/tsMuxer/bitStream.h:60:20 in BitStreamReader::getCurVal(unsigned int*)
Shadow bytes around the buggy address:
0x0c107fff8910: fa fa fa fa 00 00 00 00 00 00 00 00 00 00 00 00
0x0c107fff8920: fa fa fa fa 00 00 00 00 00 00 00 00 00 00 00 00
0x0c107fff8930: fa fa fa fa 00 00 00 00 00 00 00 00 00 00 00 00
0x0c107fff8940: fa fa fa fa 00 00 00 00 00 00 00 00 00 00 00 00
0x0c107fff8950: fa fa fa fa 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c107fff8960: fa fa fa fa 00 00 00 00 00 00 00 00 00 00 00[06]
0x0c107fff8970: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c107fff8980: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c107fff8990: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c107fff89a0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
0x0c107fff89b0: 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
==1450879==ABORTING
```

## Poc

POC.zip

Thanks for your time !

Report of the Information Security Laboratory of Ocean University of China @OUC_ISLOUC @OUC_Blue_Whale

**jcdr428** commented on Oct 16 — Collaborator

**@yangfar** the link to POC.zip is not working, can you please re-upload.

**yangfar** commented on Oct 16 — Author

Ok，I will upload POC again.

**yangfar** commented on Oct 16 — Author

Please try again.

**jcdr428** added a commit that referenced this issue on Oct 17

`Set BitStreamException` ... ✓ `a302806`

**jcdr428** commented on Oct 17 — Collaborator

**@yangfar** please try tomorrow's release, thanks.

**jcdr428** added the `bug` label on Oct 23

**jcdr428** commented on Oct 23 — Collaborator

Closing, can be reopened upon request.

**jcdr428** closed this as completed on Oct 23

Assignees

No one assigned

Labels

bug

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants