

New issue

[Jump to bottom](#)

AddressSanitizer: heap-buffer-overflow on write_node htmldoc/htmldoc/html.cxx:588 #426

✓ Closed

dramthy opened this issue on May 6, 2021 · 2 comments

Assignees



Labels

bug priority-high security

Milestone

📌 Stable

dramthy commented on May 6, 2021 • edited ▼

Hello, I found a heap-buffer-overflow in write_node

Reporter:

dramthy from Topsec Alpha Lab

test platform:

htmldoc Version : current

OS :Ubuntu 20.04.1 LTS aarch64

kernel: 5.4.0-53-generic

compiler: gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0

reproduced:

(htmldoc with asan build option)

./htmldoc-with-asan ./poc.html

[poc.zip](#)

```
=====
==25373==ERROR: AddressSanitizer: heap-buffer-overflow on address 0xfffff8680cacf at pc
0xaaaaadb308c50 bp 0xfffffe65857c0 sp 0xfffffe65857e0
READ of size 1 at 0xfffff8680cacf thread T0
    #0 0xaaaaadb308c4c in write_node /home/vm1/workspace/Projects/afl-
projects/001.html/doc/html/doc/html.cxx:588
    #1 0xaaaaadb3095d8 in write_all /home/vm1/workspace/Projects/afl-
projects/001.html/doc/html/doc/html.cxx:543
```

```

#2 0xaaaadb309630 in write_all /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/html.cxx:546
#3 0xaaaadb309630 in write_all /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/html.cxx:546
#4 0xaaaadb309630 in write_all /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/html.cxx:546
#5 0xaaaadb309630 in write_all /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/html.cxx:546
#6 0xaaaadb30a014 in write_all /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/html.cxx:538
#7 0xaaaadb30a014 in html_export /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/html.cxx:167
#8 0xaaaadb2ee52c in main /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/htmldoc.cxx:1291
#9 0xfffff8b72908c in __libc_start_main (/lib/aarch64-linux-gnu/libc.so.6+0x2408c)
#10 0xaaaadb2ee984 (/home/vm1/workspace/Projects/afl-projects/001.htmldoc/bin-with-asan-fix-
malloc2calloc+0x4b984)

```

0xfffff8680cacf is located 1 bytes to the left of 1-byte region [0xfffff8680cad0,0xfffff8680cad1) allocated by thread T0 here:

```

#0 0xfffff8befa66c in __interceptor_strdup (/lib/aarch64-linux-gnu/libasan.so.5+0x8966c)
#1 0xaaaadb35e684 in htmlReadFile /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/htmllib.cxx:796
#2 0xaaaadb30a55c in read_file /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/htmldoc.cxx:2492
#3 0xaaaadb2ee1a0 in main /home/vm1/workspace/Projects/afl-
projects/001.htmldoc/htmldoc/htmldoc/htmldoc.cxx:1177
#4 0xfffff8b72908c in __libc_start_main (/lib/aarch64-linux-gnu/libc.so.6+0x2408c)
#5 0xaaaadb2ee984 (/home/vm1/workspace/Projects/afl-projects/001.htmldoc/bin-with-asan-fix-
malloc2calloc+0x4b984)

```

SUMMARY: AddressSanitizer: heap-buffer-overflow /home/vm1/workspace/Projects/afl-projects/001.htmldoc/htmldoc/htmldoc/html.cxx:588 in write_node

Shadow bytes around the buggy address:

```

0x200ff0d01900: fa fa 00 06 fa fa 00 05 fa fa 00 06 fa fa 00 06
0x200ff0d01910: fa fa 00 04 fa fa 00 05 fa fa 00 05 fa fa 00 06
0x200ff0d01920: fa fa 00 04 fa fa 00 04 fa fa 00 05 fa fa 00 07
0x200ff0d01930: fa fa 05 fa fa fa 00 00 fa fa 07 fa fa fa 06 fa
0x200ff0d01940: fa fa 00 00 fa fa 04 fa fa fa 00 01 fa fa 05 fa
=>0x200ff0d01950: fa fa 02 fa fa fa 02 fa fa[fa]01 fa fa fa 02 fa
0x200ff0d01960: fa fa 00 05 fa fa 06 fa fa fa 03 fa fa fa 03 fa
0x200ff0d01970: fa fa 04 fa fa fa 00 02 fa fa 00 02 fa fa 00 02
0x200ff0d01980: fa fa 00 02 fa fa 00 02 fa fa 04 fa fa fa 04 fa
0x200ff0d01990: fa fa 03 fa fa fa 03 fa fa fa 03 fa fa fa 00 05
0x200ff0d019a0: fa fa 00 06 fa fa 00 05 fa fa 00 07 fa fa 00 07

```

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
==25373==ABORTING
```

this bug in `htmlldoc/htmlldoc/html.cxx:588`:

```
if (t->data[strlen((char *)t->data) - 1] == '\n')
    col = 0;
```

similar to

```
#include <string.h>
#include <iostream>
int main()
{
    const char * s = "";
    std::cout<<(int)(strlen((char *)s) - 1);
    return 0;
}
```

the index out of bounds.

  **michaelsweet** self-assigned this on May 7, 2021


  **michaelsweet** added **bug** **priority-high** **security** labels on May 7, 2021

  **michaelsweet** added this to the **Stable** milestone on May 7, 2021

michaelsweet commented on May 7, 2021

Owner

Confirmed, investigating.

 **michaelsweet** added a commit that referenced this issue on May 7, 2021



Fix a crash bug with bogus text (Issue [#426](#))

✖ ee77825

michaelrsweet commented on May 7, 2021

Owner

[master [ee77825](#)] Fix a crash bug with bogus text (Issue [#426](#))



michaelrsweet closed this as completed on May 7, 2021

Assignees



michaelrsweet

Labels

bug priority-high security

Projects

None yet

Milestone

Stable

Development

No branches or pull requests

2 participants

