

New issue

Jump to bottom

AddressSanitizer: heap-buffer-overflow in pspdf_prepare_outpages() in ps-pdf.cxx:1338:15 #413



chibataiki opened this issue on Jan 26, 2021 · 3 comments

Assignees



Labels

bug

priority-high

Milestone

Stable

chibataiki commented on Jan 26, 2021 • edited

Hello, While fuzzing htmldoc, I found a heap-buffer-overflow in the pspdf_prepare_outpages() in ps-pdf.cxx:1338:15

test platform

htmldoc Version 1.9.12 git [master 6898d0a]

OS: Ubuntu 20.04.1 LTS x86_64

kernel: 5.4.0-53-generic

compiler: clang version 10.0.0-4ubuntu1

reproduced:

```
htmldoc -f demo.pdf poc2.html
```

poc(zippped for update):

[poc2.zip](#)

```
=====
==38089==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x6150003b19f8 at pc 0x000000555985 bp 0x7ffca8f71990 sp 0x7ffca8f71988
WRITE of size 4 at 0x6150003b19f8 thread T0
#0 0x555984 in pspdf_prepare_outpages() /home/htmldoc_sani/htmldoc/ps-pdf.cxx:1338:15
#1 0x555984 in pspdf_export /home/htmldoc_sani/htmldoc/ps-pdf.cxx:901
#2 0x53c845 in main /home/htmldoc_sani/htmldoc/htmldoc.cxx:1291:3
#3 0x7f44ac7c40b2 in __libc_start_main /build/glibc-eX1tMB/glibc-2.31/csu/../csu/libc-start.c:308:16
#4 0x41f8bd in _start (/home/htmldoc_sani/htmldoc/htmldoc+0x41f8bd)
```

0x6150003b19f8 is located 0 bytes to the right of 504-byte region [0x6150003b1800,0x6150003b19f8)

allocated by thread T0 here:

```
#0 0x4ee5df in __interceptor_malloc /home/goushi/work/libfuzzer-workshop/src/llvm/projects/compiler-rt/lib/asan/asan_malloc_linux.cc:146
#1 0x5500bf in pspdf_prepare_outpages() /home/htmldoc_sani/htmldoc/ps-pdf.cxx:1258:27
#2 0x5500bf in pspdf_export /home/htmldoc_sani/htmldoc/ps-pdf.cxx:901
#3 0x53c845 in main /home/htmldoc_sani/htmldoc/htmldoc.cxx:1291:3
#4 0x7f44ac7c40b2 in __libc_start_main /build/glibc-eX1tMB/glibc-2.31/csu/../csu/libc-start.c:308:16
```

SUMMARY: AddressSanitizer: heap-buffer-overflow /home/htmldoc_sani/htmldoc/ps-pdf.cxx:1338:15 in pspdf_prepare_outpages()

Shadow bytes around the buggy address:

```
0x0c2a8006e2e0: fd fd fd fd fd fd fd fd fa fa fa fa
0x0c2a8006e2f0: fa fa fa fa fa fa fa fa fa fa fa fa
0x0c2a8006e300: 00 00 00 00 00 00 00 00 00 00 00 00
0x0c2a8006e310: 00 00 00 00 00 00 00 00 00 00 00 00
0x0c2a8006e320: 00 00 00 00 00 00 00 00 00 00 00 00
0x0c2a8006e330: 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0c2a8006e330: 00 00 00 00 00 00 00 00 00 00 00 00
0x0c2a8006e340: fa fa fa fa fa fa fa fa fa fa fa fa
0x0c2a8006e350: fa fa fa fa fa fa fa fa fa fa fa fa
0x0c2a8006e360: fa fa fa fa fa fa fa fa fa fa fa fa
0x0c2a8006e370: fa fa fa fa fa fa fa fa fa fa fa fa
0x0c2a8006e380: 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
==38089==ABORTING
```

reporter: chiba of topsec alphaslab

michaelsweet self-assigned this on Jan 26, 2021


michaelsweet added bug priority-high labels on Jan 26, 2021

michaelsweet added this to the Stable milestone on Jan 26, 2021

michaelsweet commented on Jan 26, 2021

Owner

Confirmed, investigating...

 michaelsweet added a commit that referenced this issue on Apr 1, 2021


 Fix a number-up crash bug (Issue #413)

✖ 6e8a955

michaelsweet commented on Apr 1, 2021

Owner

[master 6e8a955] Fix a number-up crash bug (Issue #413)

 michaelsweet closed this as completed on Apr 1, 2021

chibataiki commented on Feb 21

Author

CVE-2021-23165 assigned

Assignees

 michaelsweet

Labels

bug priority-high

Projects

None yet

Milestone

Stable

Development

No branches or pull requests

2 participants

