Sec Bug #79037 global buffer-overflow in `mbfl_filt_conv_big5_wchar`

Submitted: 2019-12-26 21:22 UTC **Modified:** 2020-01-21 07:15 UTC
From: reza at iseclab dot org **Assigned:** stas (profile)
Status: Closed **Package:** mbstring related
PHP Version: 7.4Git-2019-12-26 (Git) **OS:** Ubuntu 16.04
Private report: No **CVE-ID:** 2020-7060

View | Add Comment | Developer | Edit

[2019-12-26 21:22 UTC] reza at iseclab dot org

```
Description:
------------
There is an Undefined Behaviour in libmbfl at `mbfl_filt_conv_big5_wchar` function which leads to a buffer overflow on
`cp950_pua_tbl` global variable.

./php --version
PHP 7.4.2-dev (cli) (built: Dec 26 2019 15:32:36) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies


Compiled with:
CC=clang-6.0  CFLAGS="-fPIE -fsanitize=address,undefined -g" LDFLAGS="-pie" ./configure --enable-exif --enable-
mbstring

Test script:
---------------
./php -r 'mb_decode_mimeheader(file_get_contents("php://stdin"));' < global-overflow-cp950_pua_tbl.poc

You can download the poc from this repository.

https://github.com/gaintcome/fuzz-php/blob/master/poc/mbstrings/global-overflow-cp950_pua_tbl.poc



Actual result:
--------------
php-src/ext/mbstring/libmbfl/filters/mbfilter_big5.c:212:11: runtime error: index 5 out of bounds for type 'unsigned
short [5][4]'
SUMMARY: UndefinedBehaviorSanitizer: undefined-behavior php-src/ext/mbstring/libmbfl/filters/mbfilter_big5.c:212:11 in
=================================================================
==32013==ERROR: AddressSanitizer: global-buffer-overflow on address 0x000004f80a4c at pc 0x00000126177b bp
0x7fffffff9540 sp 0x7fffffff9538
READ of size 2 at 0x000004f80a4c thread T0
    #0 0x126177a in mbfl_filt_conv_big5_wchar php-src/ext/mbstring/libmbfl/filters/mbfilter_big5.c:212:11
    #1 0x13b836f in mbfl_filter_output_pipe php-src/ext/mbstring/libmbfl/mbfl/mbfl_filter_output.c:41:9
    #2 0x12fa512 in mbfl_filt_conv_qprintdec php-src/ext/mbstring/libmbfl/filters/mbfilter_qprint.c:218:4
    #3 0x13a1155 in mime_header_decoder_collector php-src/ext/mbstring/libmbfl/mbfl/mbfilter.c:2285:4
    #4 0x13a33a3 in mbfl_mime_header_decode php-src/ext/mbstring/libmbfl/mbfl/mbfilter.c:2451:3
    #5 0x11de031 in zif_mb_decode_mimeheader php-src/ext/mbstring/mbstring.c:3735:8
    #6 0x39858b1 in ZEND_DO_ICALL_SPEC_RETVAL_UNUSED_HANDLER php-src/Zend/zend_vm_execute.h:1269:2
    #7 0x30cfe4e in execute_ex php-src/Zend/zend_vm_execute.h:53611:7
    #8 0x30d2f86 in zend_execute php-src/Zend/zend_vm_execute.h:57913:2
    #9 0x2a45235 in zend_eval_stringl php-src/Zend/zend_execute_API.c:1086:4
    #10 0x2a469f0 in zend_eval_stringl_ex php-src/Zend/zend_execute_API.c:1127:11
    #11 0x2a46bd0 in zend_eval_string_ex php-src/Zend/zend_execute_API.c:1138:9
    #12 0x3cbd3aa in do_cli php-src/sapi/cli/php_cli.c:992:8
    #13 0x3cb8471 in main php-src/sapi/cli/php_cli.c:1352:18
    #14 0x7ffff619882f in __libc_start_main /build/glibc-LK5gWL/glibc-2.23/csu/../csu/libc-start.c:291
    #15 0x441d28 in _start (php-src/sapi/cli/php+0x441d28)

0x000004f80a4c is located 4 bytes to the right of global variable 'cp950_pua_tbl' defined in 'php-
src/ext/mbstring/libmbfl/filters/mbfilter_big5.c:137:23' (0x4f80a20) of size 40
SUMMARY: AddressSanitizer: global-buffer-overflow php-src/ext/mbstring/libmbfl/filters/mbfilter_big5.c:212:11 in
mbfl_filt_conv_big5_wchar
Shadow bytes around the buggy address:
  0x0000809e80f0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000809e8100: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000809e8110: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000809e8120: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000809e8130: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0000809e8140: 00 00 00 00 00 00 00 00 00[f9]f9 f9 f9 f9 f9 f9
  0x0000809e8150: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000809e8160: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000809e8170: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000809e8180: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0000809e8190: 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
==32013==ABORTING
```

Patches

Add a Patch

Pull Requests

Add a Pull Request

History

All | Comments | Changes | Git/SVN commits | Related reports

[2019-12-28 12:03 UTC] cmb@php.net

```
-Status: Open
+Status: Verified
-Package: Unknown/Other Function
+Package: mbstring related
```

[2019-12-28 12:03 UTC] cmb@php.net

Thanks for reporting!

Possible fix including a test case:
<https://gist.github.com/cmb69/951f7404fda6d71400ac63dcbe3b1463>.

[2019-12-30 04:50 UTC] reza at iseclab dot org

Thanks for the fix. I can confirm I don't see the issue after applying the patch.

[2019-12-30 15:27 UTC] cmb@php.net

```
-Assigned To:
+Assigned To: nikic
```

[2019-12-30 15:27 UTC] cmb@php.net

Thanks for verifying!

@nikic, could you please have a look?

[2019-12-30 15:44 UTC] nikic@php.net

This doesn't look right to me. The is_in_cp950_pua() check should be making sure that the table entry exists. Why doesn't it?

[2019-12-30 15:52 UTC] nikic@php.net

Okay, I see the issue. The is_in_cp950_pua() code does a check for c > 0x39, while the table entry starts at 0x40 ... which would be fine if this were decimals, but this is hex, so there's still 0x3a..0x3f between there!

[2019-12-30 15:59 UTC] nikic@php.net

Here is an alternative patch and reduced test-case: https://gist.github.com/nikic/f52bd4b3c9ab12e5cee1eb89ae13f351

[2019-12-30 17:49 UTC] cmb@php.net

Ah, much better than my patch. Can we assign to Stas?

[2019-12-30 19:04 UTC] nikic@php.net

```
-Status: Verified
+Status: Assigned
-Assigned To: nikic
+Assigned To: stas
```

[2020-01-05 01:11 UTC] reza at iseclab dot org

Do you assign a CVE to this?

[2020-01-21 05:30 UTC] stas@php.net

```
-CVE-ID:
+CVE-ID: 2020-7060
```

[2020-01-21 07:16 UTC] stas@php.net

Automatic comment on behalf of stas
Revision: http://git.php.net/?p=php-src.git;a=commit;h=2bcbc95f033c31b00595ed39f79c3a99b4ed0501
Log: Fix bug #79037 (global buffer-overflow in `mbfl_filt_conv_big5_wchar`)

[2020-01-21 07:16 UTC] stas@php.net

```
-Status: Assigned
+Status: Closed
```