New issue

Jump to bottom

libpff_item_tree_create_node 4 byte read of cached out value #61

Closed hongxuchen opened this issue on Jun 23, 2018 · 2 comments

Assignees

Labels bug

hongxuchen commented on Jun 23, 2018 • edited by joachimmetz

AddressSanitizer: heap-use-after-free at libpff_item_tree.c:816

POC files:
https://github.com/ntu-sec/pocs/blob/master/libpff-4938b7a/crashes/huaf_libpff_item_tree.c%3A816_1.input.txt
https://github.com/ntu-sec/pocs/blob/master/libpff-4938b7a/crashes/huaf_libpff_item_tree.c%3A816_2.input.txt

ASan output:
https://github.com/ntu-sec/pocs/blob/master/libpff-4938b7a/crashes/huaf_libpff_item_tree.c%3A816_1.err.SIG06
https://github.com/ntu-sec/pocs/blob/master/libpff-4938b7a/crashes/huaf_libpff_item_tree.c%3A816_2.err.SIG06

joachimmetz self-assigned this on Jun 27, 2018

joachimmetz added the needs a closer look label on Jun 27, 2018

joachimmetz commented on Jul 13, 2018 • edited Member

So it looks like libpff_item_tree_create_node was not fully refactored yet as part of #3. Which seems to cause a cache miss while reading `descriptor_index_value->parent_identifier` as a 32-bit int.

```
==25786== Invalid read of size 4
==25786==    at 0x40DBE0: libpff_item_tree_create_node (libpff_item_tree.c:779)
==25786==    by 0x40D7DC: libpff_item_tree_create_node (libpff_item_tree.c:570)
==25786==    by 0x40D7DC: libpff_item_tree_create_node (libpff_item_tree.c:570)
==25786==    by 0x40E367: libpff_item_tree_create (libpff_item_tree.c:1174)
==25786==    by 0x4043C2: libpff_file_open_read (libpff_file.c:1042)
==25786==    by 0x4039E9: libpff_file_open_file_io_handle (libpff_file.c:580)
==25786==    by 0x403748: libpff_file_open (libpff_file.c:322)
==25786==    by 0x401651: info_handle_open_input (info_handle.c:298)
==25786==    by 0x402AA9: main (pffinfo.c:284)
==25786==  Address 0x58d36a8 is 24 bytes inside a block of size 32 free'd
==25786==    at 0x4C2DD18: free (vg_replace_malloc.c:530)
==25786==    by 0x467335: libpff_index_value_free (libpff_index_value.c:125)
==25786==    by 0x443207: libfcache_cache_value_set_value (libfcache_cache_value.c:409)
==25786==    by 0x442A6F: libfcache_cache_set_value_by_index (libfcache_cache.c:549)
==25786==    by 0x44E8D8: libfdata_tree_set_node_value (libfdata_tree.c:850)
==25786==    by 0x450045: libfdata_tree_node_set_node_value (libfdata_tree_node.c:905)
==25786==    by 0x462F32: libpff_index_read_node_data (libpff_index.c:989)
==25786==    by 0x44E390: libfdata_tree_get_node_value (libfdata_tree.c:627)
==25786==    by 0x44FF55: libfdata_tree_node_get_node_value (libfdata_tree_node.c:848)
==25786==    by 0x466105: libpff_index_tree_node_get_leaf_node_by_identifier (libpff_index_tree.c:530)
==25786==    by 0x4663EB: libpff_index_tree_node_get_leaf_node_by_identifier (libpff_index_tree.c:681)
==25786==    by 0x465F51: libpff_index_tree_get_leaf_node_by_identifier (libpff_index_tree.c:436)
==25786==  Block was alloc'd at
==25786==    at 0x4C2CB6B: malloc (vg_replace_malloc.c:299)
==25786==    by 0x467212: libpff_index_value_initialize (libpff_index_value.c:62)
==25786==    by 0x462D45: libpff_index_read_node_data (libpff_index.c:893)
==25786==    by 0x44E390: libfdata_tree_get_node_value (libfdata_tree.c:627)
==25786==    by 0x45027D: libfdata_tree_node_get_number_of_sub_nodes (libfdata_tree_node.c:1015)
==25786==    by 0x40D601: libpff_item_tree_create_node (libpff_item_tree.c:475)
==25786==    by 0x40D7DC: libpff_item_tree_create_node (libpff_item_tree.c:570)
==25786==    by 0x40D7DC: libpff_item_tree_create_node (libpff_item_tree.c:570)
==25786==    by 0x40E367: libpff_item_tree_create (libpff_item_tree.c:1174)
==25786==    by 0x4043C2: libpff_file_open_read (libpff_file.c:1042)
==25786==    by 0x4039E9: libpff_file_open_file_io_handle (libpff_file.c:580)
==25786==    by 0x403748: libpff_file_open (libpff_file.c:322)
```

joachimmetz changed the title ~~AddressSanitizer: heap-use-after-free at libpff_item_tree.c:816~~ **libpff_item_tree_create_node 4 byte read of cached out value** on Jul 13, 2018

joachimmetz added bug and removed needs a closer look labels on Jul 13, 2018

joachimmetz added a commit that referenced this issue on Jul 13, 2018

`Fixed use of cached out value in libpff_item_tree_create_node #61` `effae88`

joachimmetz commented on Jul 13, 2018 Member

Addressed in `effae88`

joachimmetz closed this as completed on Jul 13, 2018

This was referenced on Jul 13, 2018

AddressSanitizer: heap-use-after-free at libpff_item_tree.c:828 #62

Closed

AddressSanitizer: heap-use-after-free at libpff_item_tree.c:841 #63

Closed

Assignees

joachimmetz

Labels

bug

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants