Bug 3392568 - buffer-overflow bug

Status: OPEN
Alias: None
Product: NASM
Component: Assembler (show other bugs)
Version: 2.15.xx
Hardware: All Linux
Importance: Medium normal
Assignee: nobody
URL:
Depends on:
Blocks:

Reported: 2019-04-23 07:00 PDT by Tao lv
Modified: 2019-04-23 07:00 PDT (History)
CC List: 4 users (show)

Obtained from: Built from git using configure

Attachments	
attached poc (5.83 KB, application/octet-stream) 2019-04-23 07:00 PDT, Tao lv	Details
Add an attachment (proposed patch, testcase, etc.)	

Note
You need to log in before you can comment on or make changes to this bug.

Tao lv 2019-04-23 07:00:04 PDT Description

Created attachment 411718 [details]
attached poc

Hi, I found a buffer-overflow at function crc64i crc64.c:185(not CVE-2019-7147
function crc64ib at crc64.c:207).

The following is the output from ASAN.

```
$ ~/vuln-fuzz/program/nasm/asan-install/bin/nasm -felf64 unique-
crash/id\:000001\,sig\:06\,src\:011515\,op\:havoc\,rep\:128
=================================================================
==69295==ERROR: AddressSanitizer: global-buffer-overflow on address 0x0000005155b0
at pc 0x0000004a8079 bp 0x7ffde8ac8c60 sp 0x7ffde8ac8c50
READ of size 8 at 0x0000005155b0 thread T0
    #0 0x4a8078 in crc64i nasmlib/crc64.c:185
    #1 0x44c7fb in pp_token_hash asm/pptok.c:499
    #2 0x436bd0 in do_directive asm/preproc.c:2218
    #3 0x449699 in pp_getline asm/preproc.c:5140
    #4 0x4088a8 in assemble_file asm/nasm.c:1549
    #5 0x404933 in main asm/nasm.c:609
    #6 0x7f8ed6c3e82f in __libc_start_main (/lib/x86_64-linux-
gnu/libc.so.6+0x2082f)
    #7 0x4022f8 in _start (/home/lt/vuln-fuzz/program/nasm/asan-
install/bin/nasm+0x4022f8)

0x0000005155b1 is located 0 bytes to the right of global variable '*.LC8' defined
in 'output/codeview.c' (0x515580) of size 49
  '*.LC8' is ascii string 'codeview: relocation for unregistered symbol: %s'
0x0000005155b0 is located 48 bytes to the left of global variable '*.LC9' defined
in 'output/codeview.c' (0x5155e0) of size 31
  '*.LC9' is ascii string 'cv8_state.source_files != NULL'
SUMMARY: AddressSanitizer: global-buffer-overflow nasmlib/crc64.c:185 crc64i
Shadow bytes around the buggy address:
  0x00008009aa60: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 f9
  0x00008009aa70: f9 f9 f9 f9 00 03 f9 f9 f9 f9 f9 04 f9 f9 f9
  0x00008009aa80: f9 f9 f9 f9 00 01 f9 f9 f9 f9 f9 00 01 f9 f9
  0x00008009aa90: f9 f9 f9 f9 06 f9 f9 f9 f9 f9 f9 00 00 00 00
  0x00008009aaa0: 00 00 00 00 00 00 00 06 f9 f9 f9 00 00 00 00
=>0x00008009aab0: 00 00 00 00 00 00[01]f9 f9 f9 f9 00 00 00 07
  0x00008009aac0: f9 f9 f9 f9 00 00 02 f9 f9 f9 f9 00 00 00 f9
  0x00008009aad0: f9 f9 f9 f9 00 00 00 00 01 f9 f9 f9 f9 f9 f9
  0x00008009aae0: 00 00 06 f9 f9 f9 f9 f9 00 00 00 00 01 f9 f9
  0x00008009aaf0: f9 f9 f9 f9 00 00 04 f9 f9 f9 f9 00 00 00 04
  0x00008009ab00: f9 f9 f9 f9 00 00 06 f9 f9 f9 f9 00 00 00 02
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
  Heap right redzone:      fb
  Freed heap region:       fd
  Stack left redzone:      f1
  Stack mid redzone:       f2
  Stack right redzone:     f3
  Stack partial redzone:   f4
  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
==69295==ABORTING
```
