Heap buffer overflow in get_le64() #394

Closed giantbranch opened this issue on Jul 24, 2020 · 1 comment

giantbranch commented on Jul 24, 2020 • edited

Author: giantbranch of NSFOCUS Security Team

What's the problem (or question)?

A heap buffer overflow read in the latest commit of the devel branch

ASAN reports:

```
==4525==ERROR: AddressSanitizer: heap-buffer-overflow on address 0x7fab6ae5cdd8 at pc 0x000000757297 bp 0x7fff2d255a60 sp 0x7fff2d255a58
READ of size 8 at 0x7fab6ae5cdd8 thread T0
    #0 0x757296 in get_le64(void const*) /src/upx-multi/src/./bele.h:182:12
    #1 0x757296 in N_BELE_RTP::LEPolicy::get64(void const*) const /src/upx-multi/src/./bele_policy.h:194:18
    #2 0x5d0419 in Packer::get_te64(void const*) const /src/upx-multi/src/./packer.h:297:65
    #3 0x5d0419 in PackLinuxElf64::unpack(OutputFile*) /src/upx-multi/src/p_lx_elf.cpp:4603:43
    #4 0x6c82b0 in Packer::doUnpack(OutputFile*) /src/upx-multi/src/packer.cpp:107:5
    #5 0x7589f8 in do_one_file(char const*, char*) /src/upx-multi/src/work.cpp:160:12
    #6 0x759f42 in do_files(int, int, char**) /src/upx-multi/src/work.cpp:271:13
    #7 0x555afd in main /src/upx-multi/src/main.cpp:1538:5
    #8 0x7fab6992783f in __libc_start_main /build/glibc-e6zv40/glibc-2.23/csu/../csu/libc-start.c:291
    #9 0x41ce98 in _start (/out/upx-multi/upx-multi+0x41ce98)

0x7fab6ae5cdd8 is located 0 bytes to the right of 132568-byte region [0x7fab6ae3c800,0x7fab6ae5cdd8)
allocated by thread T0 here:
    #0 0x49519d in malloc (/out/upx-multi/upx-multi+0x49519d)
    #1 0x5697b7 in MemBuffer::alloc(unsigned long long) /src/upx-multi/src/mem.cpp:194:42

SUMMARY: AddressSanitizer: heap-buffer-overflow /src/upx-multi/src/./bele.h:182:12 in get_le64(void const*)
Shadow bytes around the buggy address:
  0x0ff5ed5c3960: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0ff5ed5c3970: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0ff5ed5c3980: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0ff5ed5c3990: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
  0x0ff5ed5c39a0: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
=>0x0ff5ed5c39b0: 00 00 00 00 00 00 00 00 00 00 00[fa]fa fa fa fa
  0x0ff5ed5c39c0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0ff5ed5c39d0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0ff5ed5c39e0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0ff5ed5c39f0: fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa fa
  0x0ff5ed5c3a00: fa fa fa fa 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
==4525==ABORTING
```

What should have happened?

Check if the file is normal, exit if abnormal

Do you have an idea for a solution?

Add more checks

How can we reproduce the issue?

upx.out -d <poc_filename>

poc:
tests_a7c92caa967187b16a8927c29de0efee0d1f2ed5_.tar.gz

```
$ ./src/upx.out -d ./tests_a7c92caa967187b16a8927c29de0efee0d1f2ed5_.tar.gz
                       Ultimate Packer for eXecutables
                          Copyright (C) 1996 - 2020
UPX git-8d1d60  Markus Oberhumer, Laszlo Molnar & John Reiser   Jan 24th 2020

        File size         Ratio      Format      Name
```

```
   --------------------   ------   -----------   -----------
Segmentation fault
```

Please tell us details about your environment.

- UPX version used (`upx --version`):

```
upx 4.0.0-git-8d1d605b3d8c+
UCL data compression library 1.03
zlib data compression library 1.2.8
LZMA SDK version 4.43
Copyright (C) 1996-2020 Markus Franz Xaver Johannes Oberhumer
Copyright (C) 1996-2020 Laszlo Molnar
Copyright (C) 2000-2020 John F. Reiser
Copyright (C) 2002-2020 Jens Medoch
Copyright (C) 1995-2005 Jean-loup Gailly and Mark Adler
Copyright (C) 1999-2006 Igor Pavlov
UPX comes with ABSOLUTELY NO WARRANTY; for details type 'upx-multi -L'.
```

- Host Operating System and version: Ubuntu 16.04.2 LTS
- Host CPU architecture: x86_64
- Target Operating System and version: same as Host
- Target CPU architecture: same as Host

jreiser added a commit that referenced this issue on Jul 25, 2020

`Check .sh_offset and .sh_size in SHT_DYNAMIC and SHT_STRNDX` ··· ✕ `49edccd`

jreiser commented on Jul 25, 2020 Collaborator

Fixed on `devel` branch by above commit.

giantbranch closed this as completed on Jul 27, 2020

markus-oberhumer pushed a commit that referenced this issue on Aug 17

`Check .sh_offset and .sh_size in SHT_DYNAMIC and SHT_STRNDX` ··· `cc9ccdb`

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

2 participants