Bug 26931 - [nm] crash with ASAN in display_rel_file

Status: RESOLVED FIXED

Alias: None

Product: binutils
Component: binutils (show other bugs)
Version: 2.35

Importance: P2 normal
Target Milestone: ---
Assignee: Nick Clifton

URL:
Keywords:

Depends on:
Blocks:

Reported: 2020-11-22 15:39 UTC by Hao Wang
Modified: 2022-06-22 06:29 UTC (History)
CC List: 1 user (show)

See Also:
Host:
Target:
Build:
Last reconfirmed: 2020-11-23 00:00:00

Attachments	
crash test case (2.39 KB, application/x-sharedlib) 2020-11-22 15:39 UTC, Hao Wang	Details
Add an attachment (proposed patch, testcase, etc.)	View All

Note

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

Hao Wang 2020-11-22 15:39:27 UTC **Description**

Created attachment 12993 [details]
crash test case

Hello,
I found a crash in nm-new when doing fuzzing experiments. And it can be reproduced in the master branch.

I downloaded source code from git, and I built it with Ubuntu 18.04 with gcc 7.5.0 with ASAN, and the following command to build nm-new from the source:

```
CFLAGS="-O1 -fsanitize=address -g" ./configure; make clean all;
```

You can reproduce the crash with the following command:

```
nm-new --synthetic <attached file>
```

The AddressSanitizer message of the crash is:

```
==85112==ERROR: AddressSanitizer: heap-use-after-free on address 0x606000000228 at
pc 0x56518d01ceeb bp 0x7fffbdc68af0 sp 0x7fffbdc68ae0
READ of size 8 at 0x606000000228 thread T0
    #0 0x56518d01ceea in _bfd_elf_slurp_secondary_reloc_section
/home/vul337/rfuzz/psrc/bintuils-asan/bfd/elf.c:12694
    #1 0x56518d09b9a0 in bfd_elf32_slurp_reloc_table
/home/vul337/rfuzz/psrc/bintuils-asan/bfd/elfcode.h:1606
    #2 0x56518d00df5e in _bfd_elf_canonicalize_dynamic_reloc
/home/vul337/rfuzz/psrc/bintuils-asan/bfd/elf.c:8667
    #3 0x56518cfd6013 in _bfd_x86_elf_get_synthetic_symtab
/home/vul337/rfuzz/psrc/bintuils-asan/bfd/elfxx-x86.c:2111
    #4 0x56518d09637f in elf_i386_get_synthetic_symtab
/home/vul337/rfuzz/psrc/bintuils-asan/bfd/elf32-i386.c:4293
    #5 0x56518cf82cd4 in display_rel_file /home/vul337/rfuzz/psrc/bintuils-
asan/binutils/nm.c:1183
    #6 0x56518cf84470 in display_file /home/vul337/rfuzz/psrc/bintuils-
asan/binutils/nm.c:1403
    #7 0x56518cf84bed in main /home/vul337/rfuzz/psrc/bintuils-
asan/binutils/nm.c:1891
    #8 0x7f65c0e04bf6 in __libc_start_main (/lib/x86_64-linux-
gnu/libc.so.6+0x21bf6)
    #9 0x56518cf7f1c9 in _start (/home/vul337/rfuzz/psrc/bintuils-asan/binutils/nm-
new+0x9b1c9)

0x606000000228 is located 40 bytes inside of 49-byte region
[0x606000000200,0x606000000231)
freed by thread T0 here:
    #0 0x7f65c14b67a8 in __interceptor_free (/usr/lib/x86_64-linux-
gnu/libasan.so.4+0xde7a8)
    #1 0x7f65c0e1818f  (/lib/x86_64-linux-gnu/libc.so.6+0x3518f)

previously allocated by thread T0 here:
    #0 0x7f65c14b6b40 in __interceptor_malloc (/usr/lib/x86_64-linux-
gnu/libasan.so.4+0xdeb40)
    #1 0x7f65c0e17e10  (/lib/x86_64-linux-gnu/libc.so.6+0x34e10)

SUMMARY: AddressSanitizer: heap-use-after-free /home/vul337/rfuzz/psrc/bintuils-
asan/bfd/elf.c:12694 in _bfd_elf_slurp_secondary_reloc_section
```

And I can also reproduce this bug in Ubuntu 16.04, the ASAN reports a HeapOverflow bug. I checked the source code and using gdb to find the root cause, the function bfd_get_symcount in elf.c:12644 returns incorrect num and trigger a heap buffer overflow in elf.c:12690, which cause illegal memory access in a freed chunk. We can add check for the return symcount at 12644.

cvs-commit@gcc.gnu.org 2020-11-23 14:07:34 UTC **Comment 1**

The master branch has been updated by Nick Clifton <nickc@sourceware.org>:

https://sourceware.org/git/gitweb.cgi?p=binutils-gdb.git;h=f60742b2a1988d276c77d5c1011143f320d9b4cb

```
commit f60742b2a1988d276c77d5c1011143f320d9b4cb
Author: Nick Clifton <nickc@redhat.com>
Date:   Mon Nov 23 14:07:02 2020 +0000
```

Fix an illegal memory access when accessing corrupt dynamic secondary relocations.

~~PR 26931~~
* elf-bfd.h (struct elf_backend_data): Add bfd_boolean field to slurp_secondary_relocs field.
(_bfd_elf_slurp_secondary_reloc_section): Update prototype.
* elf.c (_bfd_elf_slurp_secondary_reloc_section): Add new parameter. Compute number of symbols based upon the new parameter.
* elfcode.h (elf_slurp_reloc_table): Pass dynamic as new parameter.

Nick Clifton 2020-11-23 14:08:23 UTC **Comment 2**

Hi Hao,

Thanks for reporting this bug. I have checked in a patch to fix it.

Cheers
  Nick

Hao Wang **2020-11-23 15:47:02 UTC** **Comment 3**

(In reply to Nick Clifton from comment #2)
> Hi Hao,
>
>   Thanks for reporting this bug. I have checked in a patch to fix it.
>
> Cheers
>   Nick

Hi Nick,

I have tested it, and `objdump -D` and `nm-new --synthetic` works correctly now.

Cheers
  Hao

Format For Printing - XML - Clone This Bug - Top of page