Breakme2

Wednesday, December 20, 2023 5:40 PM

Embedded System Security Analysis In-Class Activity

Winter AY 2023/2024

1 breakme2

Download breakme2 from: http://classes.csse.rose-hulman.edu/ece497/breakme2

This is similar to breakme from the ELF activities. Make this program output "You win!" only by running it in the debugger.

```
# In the first shell, this will do nothing (QEMU waits for a debugger to attach)
qemu-arm -g 1234 -L /usr/arm-linux-gnueabi ./checkpass

# In the second shell, we'll start the debugger
gdb-multiarch ./breakme2

# In the debugger, we'll connect to the running program. If you don't have the
# gef prompt, you haven't pulled the latest container
gef-remote --qemu-user --qemu-binary /share/breakme2 localhost 1234
```

Looking at the disassembly dump, I looked at what was being printed in the puts commands. A word hex gets put into r0, which corresponded to the .word found directly under main. I found the word for You win!\n conveniently under the win function (which is never called) and set $r0 to this value to make it output You win!\n

```
000104b4 <win>:
   104b4:       e92d4800        push    {fp, lr}
   104b8:       e28db004        add     fp, sp, #4
   104bc:       e3a02009        mov     r2, #9
   104c0:       e59f100c        ldr     r1, [pc, #12]   ; 104d4 <win+0x20>
   104c4:       e3a00001        mov     r0, #1
   104c8:       ebffffb7        bl      103ac <write@plt>
   104cc:       e3a00000        mov     r0, #0
   104d0:       ebffffb2        bl      103a0 <exit@plt>
   104d4:       00010598        .word   0x00010598

000104d8 <main>:
   104d8:       e92d4800        push    {fp, lr}
   104dc:       e28db004        add     fp, sp, #4
   104e0:       e24dd008        sub     sp, sp, #8
   104e4:       e50b0008        str     r0, [fp, #-8]
   104e8:       e50b100c        str     r1, [fp, #-12]
   104ec:       e59f0028        ldr     r0, [pc, #40]   ; 1051c <main+0x44>
   104f0:       ebffffa1        bl      1037c <puts@plt>
   104f4:       e59f3024        ldr     r3, [pc, #36]   ; 10520 <main+0x48>
   104f8:       e5933000        ldr     r3, [r3]
   104fc:       e1a00003        mov     r0, r3
   10500:       ebffff9a        bl      10370 <fflush@plt>
   10504:       e59f0018        ldr     r0, [pc, #24]   ; 10524 <main+0x4c>
   10508:       ebffff9b        bl      1037c <puts@plt>
   1050c:       e3a03001        mov     r3, #1
   10510:       e1a00003        mov     r0, r3
   10514:       e24bd004        sub     sp, fp, #4
   10518:       e8bd8800        pop     {fp, pc}
   1051c:       000105a4        .word   0x000105a4
   10520:       00021034        .word   0x00021034
   10524:       000105c4        .word   0x000105c4
```

```
0x21034 <stdout@@GLIBC_2.4>:    "\b\315z\377"
(remote) gef➤  x/s 0x000105c4
0x105c4:        "You lose!"
(remote) gef➤  x/s 0x00010598
0x10598:        "You win!\n"
(remote) gef➤  x/s $r0
0x105c4:        "You lose!"
(remote) gef➤  set $r0=0x00010598
(remote) gef➤  c
Continuing.
```