Resolución bomba de Manuel Hidalgo

Alumno: Juan Sánchez Rodríguez

Para resolver esta práctica he usado la herramienta gdb.

Primero hacemos un "break main" y usamos "run" para que el programa avance. Después usamos "disas" para ver todo el programa, nos saldría lo siguiente:

```
=> 0x000000000040077d <+0>: push   %rbx
   0x000000000040077e <+1>:   sub    $0xa0,%rsp
   0x0000000000400785 <+8>:   mov    %fs:0x28,%rax
   0x000000000040078e <+17>: mov    %rax,0x98(%rsp)
   0x0000000000400796 <+25>: xor    %eax,%eax
   0x0000000000400798 <+27>: lea    0x10(%rsp),%rdi
   0x000000000040079d <+32>: mov    $0x0,%esi
   0x00000000004007a2 <+37>: callq  0x4005f0 <gettimeofday@plt>
   0x00000000004007a7 <+42>: lea    0x25a(%rip),%rsi        # 0x400a08
   0x00000000004007ae <+49>: mov    $0x1,%edi
   0x00000000004007b3 <+54>: mov    $0x0,%eax
   0x00000000004007b8 <+59>: callq  0x400610 <__printf_chk@plt>
   0x00000000004007bd <+64>: lea    0x30(%rsp),%rdi
   0x00000000004007c2 <+69>: mov    0x2008a7(%rip),%rdx        # 0x601070
<stdin@@GLIBC_2.2.5>
   0x00000000004007c9 <+76>: mov    $0x64,%esi
   0x00000000004007ce <+81>: callq  0x400600 <fgets@plt>
   0x00000000004007d3 <+86>: test   %rax,%rax
   0x00000000004007d6 <+89>: je     0x4007a7 <main+42>
   0x00000000004007d8 <+91>: lea    0x200889(%rip),%rdi        # 0x601068 <password>
   0x00000000004007df <+98>:  callq  0x400727 <aquiSeModificaElPassword>
---Type <return> to continue, or q <return> to quit---
   0x00000000004007e4 <+103>:           lea    0x30(%rsp),%rdi
```

```
   0x00000000004007e9 <+108>:    mov    $0x8,%edx
   0x00000000004007ee <+113>:    lea    0x200873(%rip),%rsi        # 0x601068 <password>
   0x00000000004007f5 <+120>:    callq  0x4005d0 <strncmp@plt>
   0x00000000004007fa <+125>:    test   %eax,%eax
   0x00000000004007fc <+127>:    je     0x400803 <main+134>
   0x00000000004007fe <+129>:    callq  0x400749 <boom>
   0x0000000000400803 <+134>:    lea    0x20(%rsp),%rdi
   0x0000000000400808 <+139>:    mov    $0x0,%esi
   0x000000000040080d <+144>:    callq  0x4005f0 <gettimeofday@plt>
   0x0000000000400812 <+149>:    mov    0x20(%rsp),%rax
   0x0000000000400817 <+154>:    sub    0x10(%rsp),%rax
   0x000000000040081c <+159>:    cmp    $0x5,%rax
   0x0000000000400820 <+163>:    jle    0x400827 <main+170>
   0x0000000000400822 <+165>:    callq  0x400749 <boom>
   0x0000000000400827 <+170>:    lea    0x1f6(%rip),%rsi        # 0x400a24
   0x000000000040082e <+177>:    mov    $0x1,%edi
   0x0000000000400833 <+182>:    mov    $0x0,%eax
   0x0000000000400838 <+187>:    callq  0x400610 <__printf_chk@plt>
   0x000000000040083d <+192>:    lea    0xc(%rsp),%rsi
   0x0000000000400842 <+197>:    lea    0x1ef(%rip),%rdi        # 0x400a38
   0x0000000000400849 <+204>:    mov    $0x0,%eax
---Type <return> to continue, or q <return> to quit---
   0x000000000040084e <+209>:    callq  0x400620 <__isoc99_scanf@plt>
   0x0000000000400853 <+214>:    mov    %eax,%ebx
   0x0000000000400855 <+216>:    test   %eax,%eax
   0x0000000000400857 <+218>:    jne    0x40086a <main+237>
   0x0000000000400859 <+220>:    lea    0x1db(%rip),%rdi        # 0x400a3b
   0x0000000000400860 <+227>:    mov    $0x0,%eax
   0x0000000000400865 <+232>:    callq  0x400620 <__isoc99_scanf@plt>
   0x000000000040086a <+237>:    cmp    $0x1,%ebx
   0x000000000040086d <+240>:    jne    0x400827 <main+170>
```

```
  0x000000000040086f <+242>:    mov    0x2007eb(%rip),%edi        # 0x601060
<passcode>
  0x0000000000400875 <+248>:    callq  0x400745 <aquiSeModificaElPasscode>
  0x000000000040087a <+253>:    mov    %eax,0x2007e0(%rip)        # 0x601060
<passcode>
  0x0000000000400880 <+259>:    cmp    0xc(%rsp),%eax
  0x0000000000400884 <+263>:    je     0x40088b <main+270>
  0x0000000000400886 <+265>:    callq  0x400749 <boom>
  0x000000000040088b <+270>:    lea    0x10(%rsp),%rdi
  0x0000000000400890 <+275>:    mov    $0x0,%esi
  0x0000000000400895 <+280>:    callq  0x4005f0 <gettimeofday@plt>
  0x000000000040089a <+285>:    mov    0x10(%rsp),%rax
  0x000000000040089f <+290>:    sub    0x20(%rsp),%rax
  0x00000000004008a4 <+295>:    cmp    $0x5,%rax
---Type <return> to continue, or q <return> to quit---
  0x00000000004008a8 <+299>:    jle    0x4008af <main+306>
  0x00000000004008aa <+301>:    callq  0x400749 <boom>
  0x00000000004008af <+306>:    callq  0x400763 <defused>
```

Nos fijamos en la siguiente parte:

```
  0x00000000004007ce <+81>: callq  0x400600 <fgets@plt>
  0x00000000004007d3 <+86>: test   %rax,%rax
  0x00000000004007d6 <+89>: je     0x4007a7 <main+42>
  0x00000000004007d8 <+91>: lea    0x200889(%rip),%rdi        # 0x601068 <password>
```

Podemos ver que ahí es cuando se introduce la contraseña, que está almacenada en "0x601068". Usamos "print (char*) 0x601068" para ver la contraseña almacenada y nos responde con "$1 = 0x601068 <password> "aaaaaa\n"". Si intentamos probar con esta contraseña podemos ver que responde explotando, así que nos fijamos en lo siguiente:

```
  0x00000000004007df <+98>: callq  0x400727 <aquiSeModificaElPassword>
```

```
   0x00000000004007e4 <+103>:     lea    0x30(%rsp),%rdi
   0x00000000004007e9 <+108>:     mov    $0x8,%edx
   0x00000000004007ee <+113>:     lea    0x200873(%rip),%rsi        # 0x601068 <password>
```

Aquí podemos ver que la contraseña es modificada, así que volvemos al gdb y usamos "nexti" hasta que nos salga "Introduce la contraseña:", introducimos cualquiera y usamos "nexti" 4 veces (hasta que llama a la función donde la contraseña es modificada), y usamos nuevamente "print (char*) 0x601068" y nos responde con "$3 = 0x601068 <password> "abcdef\n"", si probamos con la contraseña "abcdef" podemos ver que efectivamente es correcta.

Para conseguir el pin borramos el break que hemos puesto antes usando "d 1" y nos fijamos en el código:

```
   0x0000000000400865 <+232>:     callq  0x400620 <__isoc99_scanf@plt>
   0x000000000040086a <+237>:     cmp    $0x1,%ebx
   0x000000000040086d <+240>:     jne    0x400827 <main+170>
   0x000000000040086f <+242>:     mov    0x2007eb(%rip),%edi        # 0x601060
<passcode>
   0x0000000000400875 <+248>:     callq  0x400745 <aquiSeModificaElPasscode>
   0x000000000040087a <+253>:     mov    %eax,0x2007e0(%rip)        # 0x601060
<passcode>
   0x0000000000400880 <+259>:     cmp    0xc(%rsp),%eax
   0x0000000000400884 <+263>:     je     0x40088b <main+270>
   0x0000000000400886 <+265>:     callq  0x400749 <boom>
```

Y creamos un breakpoint en "0x0000000000400886 <+265>:   callq 0x400749 <boom>" usando "br *main+265" y volvemos a usar "run" introducimos la contraseña obtenida anteriormente y probamos con cualquier pin, si nos fijamos en el código:

```
   0x000000000040086f <+242>:     mov    0x2007eb(%rip),%edi        # 0x601060
<passcode>
   0x0000000000400875 <+248>:     callq  0x400745 <aquiSeModificaElPasscode>
   0x000000000040087a <+253>:     mov    %eax,0x2007e0(%rip)        # 0x601060
<passcode>
```

Vemos que hace lo mismo que la anterior vez, tiene un pin que luego modifica, al tener el breakpoint después de esto ya ha sido modificado, así que usamos "x/d 0x601060" y vemos que responde con "0x601060 <passcode>: 1024" y ahora probamos introducir el pin "1024", vemos que la bomba está desactivada.