# Explanation

0x08048426 <+66>: call 0x804830c <scanf@plt> : scan input

=> 0x0804842b <+71>: cmpl $0x149a,-0x4(%ebp) : compare password with value

Value can be seen with print option “p 0x149a” which is 5274 [ref]

Reference to my piazza question : <https://piazza.com/class/l8dejyrf51k4zl/post/9>

# Debug

vagrant@ubuntu-focal:/vagrant/lab1/IOLI-crackme$ gdb crackme0x01

GNU gdb (Ubuntu 9.2-0ubuntu1~20.04.1) 9.2

Copyright (C) 2020 Free Software Foundation, Inc.

License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>

This is free software: you are free to change and redistribute it.

There is NO WARRANTY, to the extent permitted by law.

Type "show copying" and "show warranty" for details.

This GDB was configured as "x86\_64-linux-gnu".

Type "show configuration" for configuration details.

For bug reporting instructions, please see:

<http://www.gnu.org/software/gdb/bugs/>.

Find the GDB manual and other documentation resources online at:

<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".

Type "apropos word" to search for commands related to "word"...

Reading symbols from crackme0x01...

(No debugging symbols found in crackme0x01)

(gdb) r

Starting program: /vagrant/lab1/IOLI-crackme/crackme0x01

IOLI Crackme Level 0x01

Password: ^C

Program received signal SIGINT, Interrupt.

0xf7fd0b49 in \_\_kernel\_vsyscall ()

(gdb) bt

#0 0xf7fd0b49 in \_\_kernel\_vsyscall ()

#1 0xf7ec4bab in read () from /lib/i386-linux-gnu/libc.so.6

#2 0xf7e4cc73 in \_IO\_file\_underflow () from /lib/i386-linux-gnu/libc.so.6

#3 0xf7e4e020 in \_IO\_default\_uflow () from /lib/i386-linux-gnu/libc.so.6

#4 0xf7e25829 in ?? () from /lib/i386-linux-gnu/libc.so.6

#5 0xf7e23429 in scanf () from /lib/i386-linux-gnu/libc.so.6

#6 0x0804842b in main ()

(gdb) tbreak \*0x0804842b

Temporary breakpoint 1 at 0x804842b

(gdb) c

Continuing.

12345

Temporary breakpoint 1, 0x0804842b in main ()

(gdb) disas

Dump of assembler code for function main:

0x080483e4 <+0>: push %ebp

0x080483e5 <+1>: mov %esp,%ebp

0x080483e7 <+3>: sub $0x18,%esp

0x080483ea <+6>: and $0xfffffff0,%esp

0x080483ed <+9>: mov $0x0,%eax

0x080483f2 <+14>: add $0xf,%eax

0x080483f5 <+17>: add $0xf,%eax

0x080483f8 <+20>: shr $0x4,%eax

0x080483fb <+23>: shl $0x4,%eax

0x080483fe <+26>: sub %eax,%esp

0x08048400 <+28>: movl $0x8048528,(%esp)

0x08048407 <+35>: call 0x804831c <printf@plt>

0x0804840c <+40>: movl $0x8048541,(%esp)

0x08048413 <+47>: call 0x804831c <printf@plt>

0x08048418 <+52>: lea -0x4(%ebp),%eax

0x0804841b <+55>: mov %eax,0x4(%esp)

0x0804841f <+59>: movl $0x804854c,(%esp)

0x08048426 <+66>: call 0x804830c <scanf@plt>

=> 0x0804842b <+71>: cmpl $0x149a,-0x4(%ebp)

0x08048432 <+78>: je 0x8048442 <main+94>

0x08048434 <+80>: movl $0x804854f,(%esp)

0x0804843b <+87>: call 0x804831c <printf@plt>

0x08048440 <+92>: jmp 0x804844e <main+106>

0x08048442 <+94>: movl $0x8048562,(%esp)

0x08048449 <+101>: call 0x804831c <printf@plt>

0x0804844e <+106>: mov $0x0,%eax

0x08048453 <+111>: leave

0x08048454 <+112>: ret

End of assembler dump.

(gdb) x /s 0x804854c

0x804854c: "%d"

(gdb) x /d 0x149a

0x149a: Cannot access memory at address 0x149a

(gdb) x 0x149a

0x149a: <error: Cannot access memory at address 0x149a>

(gdb) p 0x149a

$1 = 5274

# CHECK PASSWORD

(gdb) c

Continuing.

Invalid Password!

[Inferior 1 (process 2383) exited normally]

(gdb) r

Starting program: /vagrant/lab1/IOLI-crackme/crackme0x01

IOLI Crackme Level 0x01

Password: 5274

Password OK :)