Do You Even String

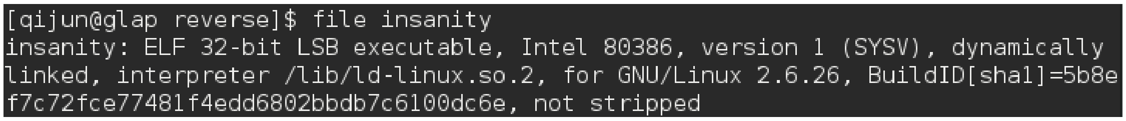
Problem: Can you find the flag in this file?

Given: insanity

Hint: Strings, Strings, more Strings

Introduction: “Do you even string” is an easy challenge designed to teach students how to perform a basic analysis against a target binary to learn more about it. Using the commands discussed in this write up should be one of your first go-tos when trying to learn more about a target executable.

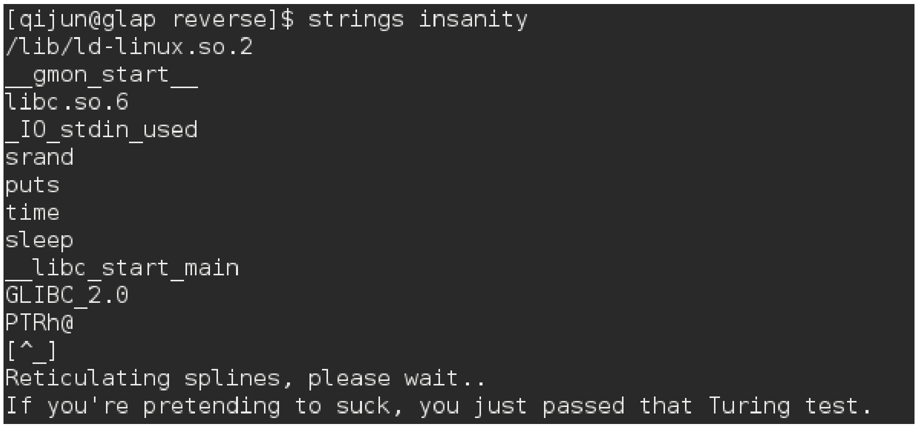
Steps:
(1) We download the binary, and begin our analysis against the file. To do this, we use two basic commands: ‘file’, and ‘strings’. The file command will provide us with basic information about the file. It’s syntax is: “file filename”.



```
[qijun@glap reverse]$ file insanity
insanity: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically
linked, interpreter /lib/ld-linux.so.2, for GNU/Linux 2.6.26, BuildID[sha1]=5b8e
f7c72fce77481f4edd6802bbdb7c6100dc6e, not stripped
```

There are two pieces of information that are important at the moment. First, this file is a 32-bit elf binary. This means that file is executable, and it was compiled on a linux system. Second, this binary is ‘not stripped’ meaning this binary was compiled with debugging info enabled. Compiling binaries with debugging info enabled allows you to read high level code inside of the debugger / disassembler. The alternate is a ‘stripped’ binary that wasn’t compiled with debugging information. This means we would only be able to read assembly / machine code in the debugger / disassembler.

(2) Now, we learn about the ‘strings’ command. Its syntax is: “strings filename”. Strings will print out all printable characters in the files you pass to it. This command is especially useful for attempting to find data that may be hidden in a non-text file, such as this binary. Go ahead and run strings against the ‘insanity’ binary.



```
[qijun@glap reverse]$ strings insanity
/lib/ld-linux.so.2
__gmon_start__
libc.so.6
_IO_stdin_used
srand
puts
time
sleep
__libc_start_main
GLIBC_2.0
PTRh@
[^_]
Reticulating splines, please wait..
If you're pretending to suck, you just passed that Turing test.
```

Can you see the flag?