Group Blue Project 1

In this lab, we used a symbolic executor, called KLEE, and a black box fuzzing tool, called Radamsa to identify errors in a program and compare each testing tool's ability to find errors. In tasks 1 and 2, we utilize radamsa to generate mutated inputs to feed into given test programs. In task 3, we utilize KLEE to run symbolic execution on the given program.

Task 1:

Question 1:

Fuzz.py identifies a crash in wisdom-alt on the first iteration.

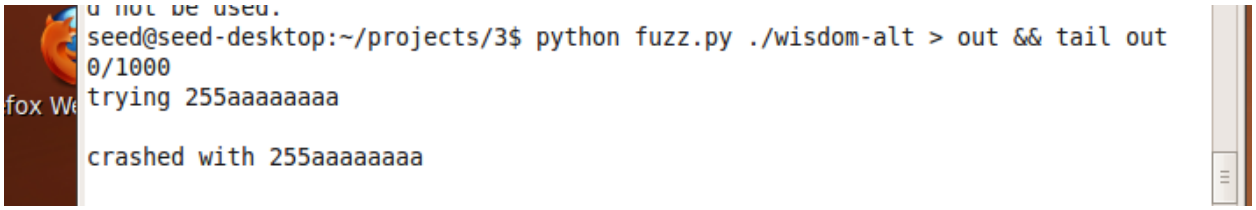

Image 1

Task 2:

Question 2:

This change allows the program to try all of the mutated inputs provided by the fuzzer. Specifically, it restricts the function pointer to point to the “get_wisdom” function (or NULL).

Question 3:

No crash is identified by the fuzzer in wisdom-alt2.

```
seed@seed-desktop:~/projects/3$ python fuzz.py ./wisdom-alt2 > out && tail out
trying 1aaaaaaaaa
1aaaaaaaa

998/1000
trying 1aaaaaaaaaaa
1aaaaaaaa

999/1000
trying
did not crash
```

Image 2

Task 3:

Question 4:

There are 2 symbolic variables set by KLEE. Their names are ‘buf’ and ‘r’, respectively.

Question 5:

The symbolic variable ‘buf’ was involved in the program. Its contents at runtime was buf=3086398448. The symbolic data was a string of null-characters or 0s depending on the interpretation of “\x00”.

```
KLEE: ERROR: /home/seed/projects/3/wisdom-alt-sym.c:60: memory error: out of bou
nd pointer
KLEE: NOTE: now ignoring this error at this location
EXITING ON ERROR:
Error: memory error: out of bound pointer
File: /home/seed/projects/3/wisdom-alt-sym.c
Line: 60
Stack:
        #0 00000116 in sym_gets (buf=3086398448) at /home/seed/projects/3/wisdom
-alt-sym.c:60
        #1 00000152 in put_wisdom () at /home/seed/projects/3/wisdom-alt-sym.c:8
0
        #2 00000276 in main (argc=1, argv=3085978392) at /home/seed/projects/3/w
isdom-alt-sym.c:123
Info:
        address: 3086398576
        next: object at 3086421600 of size 4
                MO296[4] allocated at sym_gets():  %buf_addr = alloca i8*
                    ; <i8**> [#uses=2]
        prev: object at 3086398448 of size 128
                MO160[128] allocated at put_wisdom():  %wis = alloca [128 x i8]
                        ; <[128 x i8]*> [#uses=3]

seed@seed-desktop:~/projects/3$ ls
```

Image 3

Question 6:

The klee_make_symbolic function and klee_range functions are both intrinsic functions of KLEE. The klee_make_symbolic function takes a given variable/object and turns it into an unconstrained symbolic object (marks the given variable as a symbolic input). The klee_range function ensures that the symbolic inputs that are created are within the range provided by the size and data of the given symbolic object.

```
klee-last  maze-sym.c  wisdom-alt2  wisdom-alt-sym.c
seed@seed-desktop:~/projects/3$ ktest-tool.cde klee-last/test000001.ktest
ktest file : 'klee-last/test000001.ktest'
args       : ['wisdom-alt-sym.o']
num objects: 2
object    0: name: 'buf'
object    0: size: 20
object    0: data: '\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00
\x00\x00\x00\x00\x00'
object    1: name: 'r'
object    1: size: 4
object    1: data: '\x00\x00\x00\x00'
seed@seed-desktop:~/projects/3$
```

Image 4