Fuzzer written in c and compiled using gcc.

Input Generation Strategy:

In fuzzer.c, I created a mutate function that takes in input buffer and size. It loops through the input buffer and decides whether to modify the current byte by generating a random number and comparing it to the MUTATION\_PROB threshold which I set to 0.13. If it is less than this threshold, then it will change the byte with a new randomly generated number.

In the main function, after the mutation function is called, there is a check to see if when the EXTEND\_INtERVAL of 500 is reached. When it is reached, then it is extended by EXTEND\_AMOUNT which is 10.

The mutation is then written to stdout to be piped to the test programs.

Dependencies:

Since this program is written to take a seed file in, the dependency of the mutation is dependent on this seed. The random numbers are not actually random and depend on what this seed is. This is why given the same seed and number of iterations, the output will be the same every time.

Proof of crash

Test\_programs (0-2)

Prog\_0

Text

Description automatically generated

Prog\_1

Text

Description automatically generated

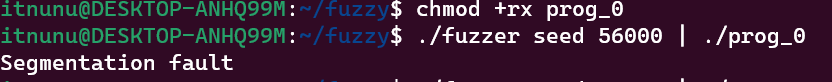
Prog\_2

Text

Description automatically generated

All\_test\_programs (0-9)

Prog\_0



Prog\_1



Prog\_2

Text

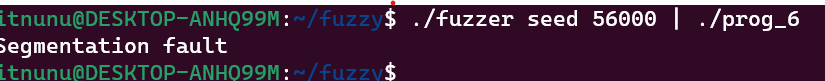
Description automatically generated with medium confidence

Program 3, 4, 5

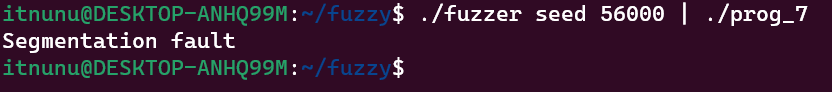
Text

Description automatically generated

Program 6



Program 7



Program 8

N/A

Program 9

