Operating Systems Lab 2 Report

Joshua Marple

September 2014

What is wrong with the original code that eventually causes it to crash?

The code crashes due to an improper memory reference. On line 3673, we see the line `the_printed_command_except_trap = the_printed_command;` which is not correct. It should be

```
the_printed_command_except_trap = savestring(the_printed_command);
```

Describe how you diagnosed the problem with the original code.

To diagnose the problem, I made substantial use of GDB. The main tool I used was definitely breakpoints. To find the problem, I let the program run once and made a note of where it segfaulted. I then put a breakpoint at that location, then executed the program again. When it hit the breakpoint, I printed out the call stack to see where the problem was at. I then worked my way up through the stack, examining the code at each point. I discovered the issue in `execute_cmd.c` by observing similar code snippets in `execute_cmd.c` and then realizing that the `savestring` call was essential to prevent improper memory access.

Describe how your solution fixes the problem. Are you confident your solution is correct?

My solution fixes the problem by fixing the memory access errors. I am confident the solution is correct only due to the proper output shown at the terminal. From a memory access perspective, it also makes sense, as I am not trying to access a nullptr anymore.