

Exercise-2: Changing a variable without directly accessing it

Introduction

In this exercise, we are expected to print “exercise succeeded” after entering some input, which particularly needs overwriting a variable in the program. This can be done by finding out if the program is vulnerable to Buffer Overflow. When we execute this program the compiler warns us for the use of **gets()**, which tells us that the input string is not checked for bounds. This fact can be used, to perform Buffer Overflow.

Source code →

```
#include <stdio.h>
#include <string.h>

int main() {

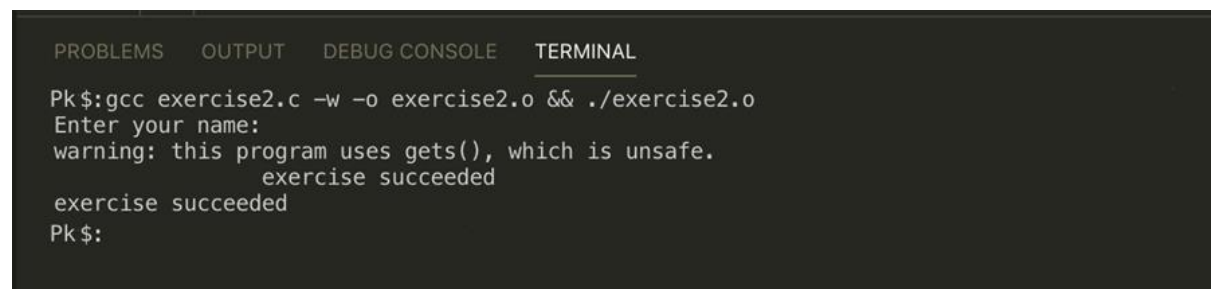
    char secret[100] = "exercise failed";
    char buf[16];

    printf("Enter your name:\n");
    gets(buf);

    printf(secret);
    printf("\n");
}
```

Buffer Overflow

The variable that is passed into **gets()** expects a string of 16 characters. Exceeding the number of characters won't give any error but will just copy the excess characters into memory that is actually assigned to some other variable.



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
Pk$: gcc exercise2.c -w -o exercise2.o && ./exercise2.o
Enter your name:
warning: this program uses gets(), which is unsafe.
exercise succeeded
exercise succeeded
Pk$:
```

Mitigation

Use of **gets()** should be avoided, as there are a lot of other functions supported in all implementations of C/C++ such as **fgets()** or **getline()**.