CSE 406 (July 2022)
Buffer Overflow Online (B - 2)

You are given the following vulnerable C program *B2.c.* Replace <param_1> and <param_2> and <param_3> in the source code with the corresponding values of Table-1.

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#define BUF_SIZE_1 <param_1>
#define BUF_SIZE_2 <param_2>
#define BUF_SIZE_3 <param_3>

int myfunc1(char* str){
    int i = 7;
    char buffer1[BUF_SIZE_1];
    strcpy(buffer1, str);
}

int myfunc2(char* str){
    char s[21] = "Hello World";
    char buffer2[BUF_SIZE_2];
    strcpy(buffer2, str);
}

int myfunc3(char* str){
    double d = 71.69;
    char buffer3[BUF_SIZE_3];
    strcpy(buffer3, str);
}

int bof(char *str){
    char buffer1[BUF_SIZE_1];
    char buffer2[BUF_SIZE_2];
    char buffer3[BUF_SIZE_3];

    int choice;
    scanf("%d",&choice);

    switch(choice){
        case 1:{
            myfunc1(str);
            break;
        }
        case 2:{
            myfunc2(str);
            break;
        }
        default:{
            myfunc3(str);
        }
    }
    return 1;
}

int main(int argc, char **argv){
    char str[517];
    FILE *badfile;

    myfunc1("Normal Execution");
    myfunc2("Normal Execution");
    myfunc3("Normal Execution");

    badfile = fopen("badfile", "r");
    if (!badfile) {
        perror("Opening badfile"); exit(1);
    }

    int length = fread(str, sizeof(char), 517, badfile);
    printf("Input size: %d\n", length);
    bof(str);
    fprintf(stdout, "==== Returned Properly ====\n");
    return 1;
}
```

Tasks:

1. First, compile the program with the 32 bit flag set as demonstrated in the class. Do not forget to turn off address space randomization and stack protection. Also, make sure that the stack is executable while compiling the program.
2. Prepare a payload (e.g. badfile) which will cause the program to open a shell with root's privilege irrespective of the user input for the variable "**choice**".
3. Rename your ***exploit.py*** file with ***1705XXX.py*** and submit on moodle.

Marks Distribution

Item	**Marks**
Task 2 (Solution for one input value)	5
Task 2 (Solution for some input values)	5
Task 2 (Solution for all input values)	5
Viva	5
Total	**20**

Table 1

Student ID	Param_1	Param_2	Param_3
1605021	143	75	63
1605036	104	69	33
1605051	120	81	33
1605085	131	84	82
1705091	116	66	220
1705092	112	61	217
1705093	133	75	31
1705094	126	61	62
1705095	121	69	133
1705096	140	69	104
1705097	110	64	157
1705098	136	72	190
1705099	108	57	63
1705100	106	57	200
1705101	131	68	54
1705102	139	97	87
1705103	140	57	32
1705104	121	63	190
1705105	122	94	50
1705106	128	84	149

Student ID	Param_1	Param_2	Param_3
1705107	110	85	81
1705108	113	87	109
1705109	112	95	142
1705110	117	84	221
1705111	134	97	209
1705112	123	84	52
1705113	105	76	133
1705114	133	59	113
1705115	116	59	108
1705116	124	60	43
1705117	124	96	102
1705118	110	95	166
1705119	135	60	190
1705120	133	57	146