

(The survey began with an informed consent statement and experiment explanation, omitted here for brevity.)

What's your age?

- ☐ Under 18+
- ☐ 18 - 24
- ☐ 25 - 34
- ☐ 35 - 44
- ☐ 45 - 54
- ☐ 55 - 64
- ☐ 65 - 74
- ☐ 75 - 84
- ☐ 85 or older

What is your gender?

- ☐ Male
- ☐ Female
- ☐ Non-binary / third gender
- ☐ I prefer not to say

I have experience with: (choice all options apply)

- ☐ Programming in C/C++
- ☐ Programming in Assembly
- ☐ Managing memory by malloc/free
- ☐ Participation in bug-hunting programs
- ☐ Participation in Capture the Flag (CTF)
- ☐ Working in a software security group in my company/university
- ☐ Exploitation stack/heap overflow in the real world
- ☐ None of the above

Do you have experience with the following symbolic execution tool?

- ☐ KLEE <http://klee.github.io/>
- ☐ angr <https://angr.io/>
- ☐ triton <https://triton-library.github.io/>
- ☐ KEY <https://www.key-project.org/>
- ☐ None of the above

Which IDE / tools / software are you familiar with when you review / debug source codes?

We may pre-install the common choices in experiment environment.

For the C code snippet below, it may contain a memory-related vulnerability.

```
#define MAX_INPUT_LEN 32
//copy user-control input string to output string, replace & with '&';
char * escaped_copy_input(char *s){
if (strlen(s) > MAX_INPUT_LEN)
    return NULL; //input too long
char *dst = (char*)malloc(MAX_INPUT_LEN * 4 + 1);
size_t dst_index = 0;
for (size_t i = 0; i < strlen(s); i++ )
    if( '&' == s[i] ){
        dst[dst_index++] = '&';
        dst[dst_index++] = 'a';
        dst[dst_index++] = 'm';
        dst[dst_index++] = 'p';
        dst[dst_index++] = '.';
    }
    else
        dst[dst_index++] = s[i];
dst[dst_index] = '\0';
return dst;
}
```

Do you find any bugs in the source code?

☐ Yes

☐ No

If possible, please briefly describe how the bug occurs by code line number and vulnerability type?

If you have successfully identified the bug, what is value of the parameter (char*s) that triggers it?

(The survey ended with optional contact questions, omitted here for brevity.)