

Problem Sheet #3

Problem 3.1: *fizzbuzz control flow integrity*

(7+2+1 = 10 points)

A seminal paper on control flow integrity is the paper by Martín Abadi, Mihai Budiu, Úlfar Erlingson and Jay Ligatti titled "[Control-Flow Integrity Principles, Implementations, and Applications](#)", published in the ACM Transactions on Information and System, 13(1), 2009.

- a) The paper describes how labels can be used to protect indirect jumps. Draw a figure similar to Fig. 1 in the paper for the following C program.

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

static int show(char *s)
{
    return printf("%s", s);
}

static int guarded_show(char *(fp)(int n), int n)
{
    char *s = fp ? fp(n) : NULL;
    if (s) {
        return show(s);
    }
    return 0;
}

static char* fizz(int n)
{
    return (n % 3 == 0) ? "Fizz" : NULL;
}

static char* buzz(int n)
{
    return (n % 5 == 0) ? "Buzz" : NULL;
}

static void handle(char *s)
{
    char* (*fp[])(int) = { fizz, buzz, NULL };
    int n = atoi(s);
    if (n > 0) {
        int cnt = 0;
        for (int i = 0; fp[i] != NULL; i++) {
            cnt += guarded_show(fp[i], n);
        }
        if (cnt) {
            return;
        }
    }
    show(s);
}

int main(int argc, char *argv[])
{
    for (int i = 1; i < argc; i++) {
        char *separator = (i < argc-1) ? " " : "\n";
```

```
        handle(argv[i]);
        show(separator);
    }
    return EXIT_SUCCESS;
}
```

b) The paper provides a formal definition of a control flow graph (CFG). The requirement (6) is stated as follows:

(6) if $w_0, w_1 \in \text{dom}(M_c)$, then $\text{succ}(w_0) \cap \text{succ}(w_1) = \emptyset$ or $\text{succ}(w_0) = \text{succ}(w_1)$.

Describe the meaning of this requirement in your own words.

c) What are the NXD and NWC assumptions? How can these two assumptions be justified?