

# Typographical Error

## Description

A typographical error can occur for example when the intent of a defined operation is to sum a number to a variable (+) but it has accidentally been defined in a wrong way (=+), introducing a typo which happens to be a valid operator. Instead of calculating the sum it initializes the variable again.

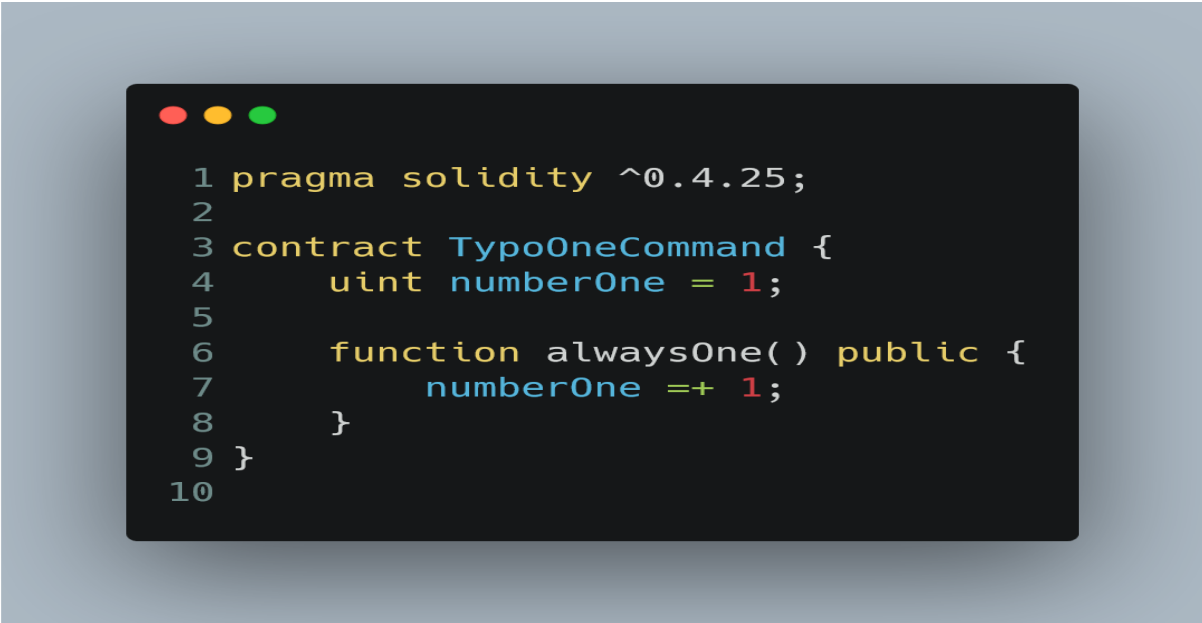
The unary + operator is deprecated in new solidity compiler versions.

## Remediation

The weakness can be avoided by performing pre-condition checks on any math operation or using a vetted library for arithmetic calculations such as SafeMath developed by OpenZeppelin.

## Example:

*Code:*

A screenshot of a code editor window with a dark background and light-colored text. The code is Solidity and contains a typographical error. The code is as follows:

```
1 pragma solidity ^0.4.25;
2
3 contract TypoOneCommand {
4     uint numberOne = 1;
5
6     function alwaysOne() public {
7         numberOne =+ 1;
8     }
9 }
10
```

The error is on line 7, where the operator is written as "=+" instead of "+".

*Explanation:*

Instead of adding value to the variable, the value will be assigned to the variable in line 7. This is due to the typographical mistake, which causes a significant computation inaccuracy.