# CWE Detail – CWE-480

## Description

The product accidentally uses the wrong operator, which changes the logic in security-relevant ways.

## Extended Description

These types of errors are generally the result of a typo by the programmer.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2022-3979: Chain: data visualization program written in PHP uses the "!=" operator instead of the type-strict "!==" operator (CWE-480) when validating hash values, potentially leading to an incorrect type conversion (CWE-704)

**•** CVE-2021-3116: Chain: Python-based HTTP Proxy server uses the wrong boolean operators (CWE-480) causing an incorrect comparison (CWE-697) that identifies an authN failure if all three conditions are met instead of only one, allowing bypass of the proxy authentication (CWE-1390)

## Modes of Introduction

**•** Implementation: N/A

## Common Consequences

**•** Impact: Alter Execution Logic — Notes: This weakness can cause unintended logic to be executed and other unexpected application behavior.

## Applicable Platforms

**•** C (Class: None, Prevalence: Sometimes)

**•** C++ (Class: None, Prevalence: Sometimes)

**•** Perl (Class: None, Prevalence: Sometimes)

**•** None (Class: Not Language-Specific, Prevalence: Undetermined)

## Demonstrative Examples

**•** However, the expression to be evaluated in the if statement uses the assignment operator "=" rather than the comparison operator "==". The result of using the assignment operator instead of the comparison operator causes the int variable to be reassigned locally and the expression in the if statement will always evaluate to the value on the right hand side of the expression. This will result in the input value not being properly validated, which can cause unexpected results.

**•** The push method includes an expression to assign the integer value to the location in the stack pointed to by the pointer variable.

**•** The following code [REF-1377] illustrates an instance of a vulnerable implementation of debug mode. The core correctly checks if the debug requests have sufficient privileges and enables the debug\_mode\_d and debug\_mode\_q signals. It also correctly checks for debug password and enables umode\_i signal.