# CWE Detail – CWE-733

## Description

The developer builds a security-critical protection mechanism into the software, but the compiler optimizes the program such that the mechanism is removed or modified.

## Extended Description

N/A

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2008-1685: C compiler optimization, as allowed by specifications, removes code that is used to perform checks to detect integer overflows.

**•** CVE-2019-1010006: Chain: compiler optimization (CWE-733) removes or modifies code used to detect integer overflow (CWE-190), allowing out-of-bounds write (CWE-787).

## Related Attack Patterns (CAPEC)

* CAPEC-10
* CAPEC-24
* CAPEC-46
* CAPEC-8
* CAPEC-9

## Common Consequences

**•** Impact: Bypass Protection Mechanism, Other — Notes:

## Applicable Platforms

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

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

**•** None (Class: Compiled, Prevalence: Undetermined)

## Demonstrative Examples

**•** The code in the example will behave correctly if it is executed verbatim, but if the code is compiled using an optimizing compiler, such as Microsoft Visual C++ .NET or GCC 3.x, then the call to memset() will be removed as a dead store because the buffer pwd is not used after its value is overwritten [18]. Because the buffer pwd contains a sensitive value, the application may be vulnerable to attack if the data are left memory resident. If attackers are able to access the correct region of memory, they may use the recovered password to gain control of the system.