# CWE Detail – CWE-500

## Description

An object contains a public static field that is not marked final, which might allow it to be modified in unexpected ways.

## Extended Description

Public static variables can be read without an accessor and changed without a mutator by any classes in the application.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Modes of Introduction

**•** Implementation: N/A

## Common Consequences

**•** Impact: Modify Application Data — Notes: The object could potentially be tampered with.

**•** Impact: Read Application Data — Notes: The object could potentially allow the object to be read.

## Potential Mitigations

**•** Architecture and Design: Clearly identify the scope for all critical data elements, including whether they should be regarded as static. (Effectiveness: N/A)

**•** Implementation: Make any static fields private and constant. A constant field is denoted by the keyword 'const' in C/C++ and ' final' in Java (Effectiveness: N/A)

## Applicable Platforms

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

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

## Demonstrative Examples

**•** Having a public static variable that is not marked final (constant) may allow the variable to the altered in a way not intended by the application. In this example the String variable can be modified to indicate a different on nonexistent properties file which could cause the application to crash or caused unexpected behavior.