# CWE Detail – CWE-413

## Description

The product does not lock or does not correctly lock a resource when the product must have exclusive access to the resource.

## Extended Description

When a resource is not properly locked, an attacker could modify the resource while it is being operated on by the product. This might violate the product's assumption that the resource will not change, potentially leading to unexpected behaviors.

## Threat-Mapped Scoring

Score: 1.8

Priority: P4 - Informational (Low)

## Observed Examples (CVEs)

**•** CVE-2022-20141: Chain: an operating system kernel has insufficent resource locking (CWE-413) leading to a use after free (CWE-416).

## Modes of Introduction

**•** Architecture and Design: N/A

**•** Implementation: N/A

## Common Consequences

**•** Impact: Modify Application Data, DoS: Instability, DoS: Crash, Exit, or Restart — Notes:

## Potential Mitigations

**•** Architecture and Design: Use a non-conflicting privilege scheme. (Effectiveness: N/A)

**•** Architecture and Design: Use synchronization when locking a resource. (Effectiveness: N/A)

## Applicable Platforms

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

## Demonstrative Examples

**•** However, the code does not check the value returned by pthread\_mutex\_lock() for errors. If pthread\_mutex\_lock() cannot acquire the mutex for any reason, the function may introduce a race condition into the program and result in undefined behavior.

**•** However, the deposit and withdraw methods have shared access to the account balance private class variable. This can result in a race condition if multiple threads attempt to call the deposit and withdraw methods simultaneously where the account balance is modified by one thread before another thread has completed modifying the account balance. For example, if a thread attempts to withdraw funds using the withdraw method before another thread that is depositing funds using the deposit method completes the deposit then there may not be sufficient funds for the withdraw transaction.