# CWE Detail – CWE-1342

## Description

The processor does not properly clear microarchitectural state after incorrect microcode assists or speculative execution, resulting in transient execution.

## Extended Description

In many processor architectures an exception, mis-speculation, or microcode assist results in a flush operation to clear results that are no longer required. This action prevents these results from influencing architectural state that is intended to be visible from software. However, traces of this transient execution may remain in microarchitectural buffers, resulting in a change in microarchitectural state that can expose sensitive information to an attacker using side-channel analysis. For example, Load Value Injection (LVI) [REF-1202] can exploit direct injection of erroneous values into intermediate load and store buffers. Several conditions may need to be fulfilled for a successful attack: incorrect transient execution that results in remanence of sensitive information; attacker has the ability to provoke microarchitectural exceptions; operations and structures in victim code that can be exploited must be identified.

## Threat-Mapped Scoring

Score: 3.75

Priority: P2 - Serious (High)

## Observed Examples (CVEs)

**•** CVE-2020-0551: Load value injection in some processors utilizing speculative execution may allow an authenticated user to enable information disclosure via a side-channel with local access.

## Related Attack Patterns (CAPEC)

* CAPEC-696

## Modes of Introduction

**•** Architecture and Design: N/A

**•** Requirements: N/A

## Common Consequences

**•** Impact: Modify Memory, Read Memory, Execute Unauthorized Code or Commands — Notes:

## Potential Mitigations

**•** Architecture and Design: Hardware ensures that no illegal data flows from faulting micro-ops exists at the microarchitectural level. (Effectiveness: High)

**•** Build and Compilation: Include instructions that explicitly remove traces of unneeded computations from software interactions with microarchitectural elements e.g. lfence, sfence, mfence, clflush. (Effectiveness: High)

## Applicable Platforms

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

## Demonstrative Examples

**•** Consider the code gadget:

## Notes

**•** Relationship: CWE-1342 differs from CWE-1303, which is related to misprediction and biasing microarchitectural components, while CWE-1342 addresses illegal data flows and retention. For example, Spectre is an instance of CWE-1303 biasing branch prediction to steer the transient execution indirectly.

**•** Maintenance: As of CWE 4.9, members of the CWE Hardware SIG are closely analyzing this entry and others to improve CWE's coverage of transient execution weaknesses, which include issues related to Spectre, Meltdown, and other attacks. Additional investigation may include other weaknesses related to microarchitectural state. As a result, this entry might change significantly in CWE 4.10.