# CWE Detail – CWE-1313

## Description

During runtime, the hardware allows for test or debug logic (feature) to be activated, which allows for changing the state of the hardware. This feature can alter the intended behavior of the system and allow for alteration and leakage of sensitive data by an adversary.

## Extended Description

An adversary can take advantage of test or debug logic that is made accessible through the hardware during normal operation to modify the intended behavior of the system. For example, an accessible Test/debug mode may allow read/write access to any system data. Using error injection (a common test/debug feature) during a transmit/receive operation on a bus, data may be modified to produce an unintended message. Similarly, confidentiality could be compromised by such features allowing access to secrets.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Observed Examples (CVEs)

**•** CVE-2021-33150: Hardware processor allows activation of test or debug logic at runtime.

**•** CVE-2021-0146: Processor allows the activation of test or debug logic at runtime, allowing escalation of privileges

## Related Attack Patterns (CAPEC)

* CAPEC-121

## Modes of Introduction

**•** Architecture and Design: Such issues could be introduced during hardware architecture and design and identified later during Testing or System Configuration phases.

**•** Implementation: Such issues could be introduced during implementation and identified later during Testing or System Configuration phases.

**•** Integration: Such issues could be introduced during integration and identified later during Testing or System configuration phases.

## Common Consequences

**•** Impact: Modify Memory, Read Memory, DoS: Crash, Exit, or Restart, DoS: Instability, DoS: Resource Consumption (CPU), DoS: Resource Consumption (Memory), DoS: Resource Consumption (Other), Execute Unauthorized Code or Commands, Gain Privileges or Assume Identity, Bypass Protection Mechanism, Alter Execution Logic, Quality Degradation, Unexpected State, Reduce Performance, Reduce Reliability — Notes:

## Potential Mitigations

**•** Architecture and Design: Insert restrictions on when the hardware's test or debug features can be activated. For example, during normal operating modes, the hardware's privileged modes that allow access to such features cannot be activated. Configuring the hardware to only enter a test or debug mode within a window of opportunity such as during boot or configuration stage. The result is disablement of such test/debug features and associated modes during normal runtime operations. (Effectiveness: N/A)

**•** Implementation: Insert restrictions on when the hardware's test or debug features can be activated. For example, during normal operating modes, the hardware's privileged modes that allow access to such features cannot be activated. Configuring the hardware to only enter a test or debug mode within a window of opportunity such as during boot or configuration stage. The result is disablement of such test/debug features and associated modes during normal runtime operations. (Effectiveness: N/A)

**•** Integration: Insert restrictions on when the hardware's test or debug features can be activated. For example, during normal operating modes, the hardware's privileged modes that allow access to such features cannot be activated. Configuring the hardware to only enter a test or debug mode within a window of opportunity such as during boot or configuration stage. The result is disablement of such test/debug features and associated modes during normal runtime operations. (Effectiveness: N/A)

## Applicable Platforms

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