# CWE Detail – CWE-1304

## Description

The product performs a power save/restore  
 operation, but it does not ensure that the integrity of  
 the configuration state is maintained and/or verified between  
 the beginning and ending of the operation.

## Extended Description

Before powering down, the Intellectual  
 Property (IP) saves current state (S) to persistent  
 storage such as flash or always-on memory in order to  
 optimize the restore operation. During this process,  
 an attacker with access to the persistent storage may  
 alter (S) to a configuration that could potentially  
 modify privileges, disable protections, and/or cause  
 damage to the hardware. If the IP does not validate  
 the configuration state stored in persistent memory,  
 upon regaining power or becoming operational again,  
 the IP could be compromised through the activation of  
 an unwanted/harmful configuration.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Related Attack Patterns (CAPEC)

* CAPEC-176

## Modes of Introduction

**•** Architecture and Design: Weakness introduced via missing internal integrity guarantees during power save/restore

**•** Integration: Weakness introduced via missing external integrity verification during power save/restore

## Common Consequences

**•** Impact: DoS: Instability, DoS: Crash, Exit, or Restart, DoS: Resource Consumption (Other), Gain Privileges or Assume Identity, Bypass Protection Mechanism, Alter Execution Logic, Quality Degradation, Unexpected State, Reduce Maintainability, Reduce Performance, Reduce Reliability — Notes:

## Potential Mitigations

**•** Architecture and Design: Inside the IP, incorporate integrity checking  
 on the configuration state via a cryptographic  
 hash. The hash can be protected inside the IP such as  
 by storing it in internal registers which never lose  
 power. Before powering down, the IP performs a hash of  
 the configuration and saves it in these persistent  
 registers. Upon restore, the IP performs a hash of the  
 saved configuration and compares it with the  
 saved hash. If they do not match, then the IP should  
 not trust the configuration. (Effectiveness: N/A)

**•** Integration: Outside the IP, incorporate integrity checking  
 of the configuration state via a trusted agent. Before  
 powering down, the trusted agent performs a hash of the  
 configuration and saves the hash in persistent storage.  
 Upon restore, the IP requests the trusted agent  
 validate its current configuration. If the  
 configuration hash is invalid, then the IP should not  
 trust the configuration. (Effectiveness: N/A)

**•** Integration: Outside the IP, incorporate a protected  
 environment that prevents undetected modification of  
 the configuration state by untrusted agents. Before  
 powering down, a trusted agent saves the IP's  
 configuration state in this protected location that  
 only it is privileged to. Upon restore, the trusted  
 agent loads the saved state into the IP. (Effectiveness: N/A)

## Applicable Platforms

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

## Demonstrative Examples

**•** The following pseudo-code is the proper workflow for the integrity checking mitigation: