# CWE Detail – CWE-1270

## Description

The product implements a Security Token mechanism to differentiate what actions are allowed or disallowed when a transaction originates from an entity. However, the Security Tokens generated in the system are incorrect.

## Extended Description

Systems-On-a-Chip (SoC) (Integrated circuits and hardware engines) implement Security Tokens to differentiate and identify actions originated from various agents. These actions could be "read", "write", "program", "reset", "fetch", "compute", etc. Security Tokens are generated and assigned to every agent on the SoC that is either capable of generating an action or receiving an action from another agent. Every agent could be assigned a unique, Security Token based on its trust level or privileges. Incorrectly generated Security Tokens could result in the same token used for multiple agents or multiple tokens being used for the same agent. This condition could result in a Denial-of-Service (DoS) or the execution of an action that in turn could result in privilege escalation or unintended access.

## Threat-Mapped Scoring

Score: 1.9

Priority: P3 - Important (Medium)

## Related Attack Patterns (CAPEC)

* CAPEC-121
* CAPEC-633
* CAPEC-681

## Attack TTPs

**•** T1134: Access Token Manipulation (Tactics: defense-evasion, privilege-escalation)

## Modes of Introduction

**•** Architecture and Design: N/A

**•** Implementation: N/A

## Common Consequences

**•** Impact: Modify Files or Directories, Execute Unauthorized Code or Commands, Bypass Protection Mechanism, Gain Privileges or Assume Identity, Read Memory, Modify Memory, DoS: Crash, Exit, or Restart — Notes:

## Potential Mitigations

**•** Architecture and Design: Generation of Security Tokens should be reviewed for design inconsistency and common weaknesses. Security-Token definition and programming flow should be tested in pre-silicon and post-silicon testing. (Effectiveness: N/A)

## Applicable Platforms

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

## Demonstrative Examples

**•** Assume the system has two agents: a Main-controller and an Aux-controller. The respective Security Tokens are "1" and "2". Register Description Default AES\_ENC\_DEC\_KEY\_0 AES key [0:31] for encryption or decryption 0x00000000 AES\_ENC\_DEC\_KEY\_1 AES key [32:63] for encryption or decryption 0x00000000 AES\_ENC\_DEC\_KEY\_2 AES key [64:95] for encryption or decryption 0x00000000 AES\_ENC\_DEC\_KEY\_3 AES key [96:127] for encryption or decryption 0x00000000 AES\_KEY\_ACCESS\_POLICY AES key access register [31:0] 0x00000002