# CWE Detail – CWE-1431

## Description

The product uses a hardware module implementing a cryptographic
 algorithm that writes sensitive information about the intermediate
 state or results of its cryptographic operations via one of its output
 wires (typically the output port containing the final result).

## Extended Description

N/A

## Threat-Mapped Scoring

Score: 3.0

Priority: P2 - Serious (High)

## Modes of Introduction

**•** Implementation: This can occur when intermediate cryptographic states are
 directly assigned to output wires or ports.

## Common Consequences

**•** Impact: Read Memory, Read Application Data — Notes:

## Potential Mitigations

**•** Architecture and Design: Designers/developers
 should add or modify existing control flow
 logic along any data flow paths that
 connect "sources" (signals with
 intermediate cryptographic state/results)
 with "sinks" (hardware module outputs and
 other signals outside of trusted
 cryptographic zone). The control flow
 logic should only allow cryptographic
 results to be driven to "sinks" when
 appropriate conditions are satisfied
 (typically when the final result for a
 cryptographic operation has been
 generated). When the appropriate
 conditions are not satisfied (i.e., before
 or during a cryptographic operation), the
 control flow logic should drive a safe
 default value to
 "sinks". (Effectiveness: High)

**•** Implementation: Designers/developers
 should add or modify existing control flow
 logic along any data flow paths that
 connect "sources" (signals with
 intermediate cryptographic state/results)
 with "sinks" (hardware module outputs and
 other signals outside of trusted
 cryptographic zone). The control flow
 logic should only allow cryptographic
 results to be driven to "sinks" when
 appropriate conditions are satisfied
 (typically when the final result for a
 cryptographic operation has been
 generated). When the appropriate
 conditions are not satisfied (i.e., before
 or during a cryptographic operation), the
 control flow logic should drive a safe
 default value to
 "sinks". (Effectiveness: High)

## Applicable Platforms

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

## Demonstrative Examples

**•** In line 50 above, data\_state\_q is assigned to data\_o. Since data\_state\_q
 contains intermediate state/results, this allows an attacker to obtain
 these results through data\_o.