# CWE Detail – CWE-354

## Description

The product does not validate or incorrectly validates the integrity check values or "checksums" of a message. This may prevent it from detecting if the data has been modified or corrupted in transmission.

## Extended Description

Improper validation of checksums before use results in an unnecessary risk that can easily be mitigated. The protocol specification describes the algorithm used for calculating the checksum. It is then a simple matter of implementing the calculation and verifying that the calculated checksum and the received checksum match. Improper verification of the calculated checksum and the received checksum can lead to far greater consequences.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Related Attack Patterns (CAPEC)

* CAPEC-145
* CAPEC-463
* CAPEC-75

## Modes of Introduction

**•** Architecture and Design: N/A

**•** Implementation: REALIZATION: This weakness is caused during implementation of an architectural security tactic.

## Common Consequences

**•** Impact: Modify Application Data, Other — Notes: Integrity checks usually use a secret key that helps authenticate the data origin. Skipping integrity checking generally opens up the possibility that new data from an invalid source can be injected.

**•** Impact: Other — Notes: Data that is parsed and used may be corrupted.

**•** Impact: Hide Activities, Other — Notes: Without a checksum check, it is impossible to determine if any changes have been made to the data after it was sent.

## Potential Mitigations

**•** Implementation: Ensure that the checksums present in messages are properly checked in accordance with the protocol specification before they are parsed and used. (Effectiveness: N/A)

## Applicable Platforms

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

## Demonstrative Examples

**•** N/A