# CWE Detail – CWE-695

## Description

The product uses low-level functionality that is explicitly prohibited by the framework or specification under which the product is supposed to operate.

## Extended Description

The use of low-level functionality can violate the specification in unexpected ways that effectively disable built-in protection mechanisms, introduce exploitable inconsistencies, or otherwise expose the functionality to attack.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Related Attack Patterns (CAPEC)

* CAPEC-36

## Modes of Introduction

**•** Implementation: N/A

## Common Consequences

**•** Impact: Other — Notes:

## Demonstrative Examples

**•** Because the example is implemented in Java, it may appear that it is immune to memory issues like buffer overflow vulnerabilities. Although Java does do a good job of making memory operations safe, this protection does not extend to vulnerabilities occurring in source code written in other languages that are accessed using the Java Native Interface. Despite the memory protections offered in Java, the C code in this example is vulnerable to a buffer overflow because it makes use of gets(), which does not check the length of its input.

**•** A Socket object is created directly within the Java servlet, which is a dangerous way to manage remote connections.