# CWE Detail – CWE-1310

## Description

Missing an ability to patch ROM code may leave a System or System-on-Chip (SoC) in a vulnerable state.

## Extended Description

A System or System-on-Chip (SoC) that implements a boot process utilizing security mechanisms such as Root-of-Trust (RoT) typically starts by executing code from a Read-only-Memory (ROM) component. The code in ROM is immutable, hence any security vulnerabilities discovered in the ROM code can never be fixed for the systems that are already in use. A common weakness is that the ROM does not have the ability to patch if security vulnerabilities are uncovered after the system gets shipped. This leaves the system in a vulnerable state where an adversary can compromise the SoC.

## Threat-Mapped Scoring

Score: 0.0

Priority: Unclassified

## Related Attack Patterns (CAPEC)

* CAPEC-682

## Modes of Introduction

**•** Architecture and Design: This issue could be introduced during hardware architecture and design and can be identified later during Testing.

**•** Implementation: This issue could be introduced during implementation and can be identified later during Testing.

**•** Integration: This issue could be introduced during integration and can be identified later during Testing.

**•** Manufacturing: This issue could be introduced during manufacturing and can be identified later during Testing.

## Common Consequences

**•** Impact: Varies by Context, Reduce Maintainability — Notes: When the system is unable to be patched, it can be left in a vulnerable state.

## Potential Mitigations

**•** Architecture and Design: Secure patch support to allow ROM code to be patched on the next boot. (Effectiveness: Moderate)

**•** Architecture and Design: Support patches that can be programmed in-field or during manufacturing through hardware fuses. This feature can be used for limited patching of devices after shipping, or for the next batch of silicon devices manufactured, without changing the full device ROM. (Effectiveness: Moderate)

## Applicable Platforms

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

## Demonstrative Examples

**•** ROM does not have built-in application-programming interfaces (APIs) to patch if the code is vulnerable. Implement mechanisms to patch the vulnerable ROM code.

**•** The above implementation causes the ROM data to be hardcoded for the linux system (rom\_rdata\_linux) regardless of the value of ariane\_boot\_sel\_i. Therefore, the data (rom\_rdata\_patch) from the patchable ROM code is never used [REF-1396]. This weakness disables the ROM's ability to be patched. If attackers uncover security vulnerabilities in the ROM, the users must replace the entire device. Otherwise, the weakness exposes the system to a vulnerable state forever. A fix to this issue is to enable rom\_rdata to be selected from the patchable rom (rom\_rdata\_patch) [REF-1397].