Platform Firmware Resiliency

Presenter: Moung Heum Eum – Senior Embedded Solutions Engineer

MICROCHIP

Abstract

- Does your system boot from an SPI Flash?
- How do you protect your system from risk during the boot process and during firmware updates?
- How does your system detect and recover from corrupted flash files?
- We will discuss the NIST 800-193 guidelines for Platform Firmware Resiliency today

System Resilience

- **<u>System Resilience:</u>** ***The capability of a system with specific characteristics before, during and after a disruption to absorb the disruption, recover to an acceptable level of performance and sustain that level for an acceptable period of time***
  - International Council of Systems Engineering (INCOSE)

- **<u>Cyber Resilience:</u> Ability to anticipate, withstand, recover from and adapt to adverse conditions, stresses, attacks, or compromises on systems that include cyber resources**

The Platform

- **Comprised of hardware and firmware necessary to initialize components and boot the system to a point software, or an operating system, can be loaded**
- **Collection of devices that provide the functional capabilities and services needed by the operating systems and applications, examples:**
  - Embedded Controllers (EC)
  - Trusted Platform Module (TPM)
  - Baseboard Management Controller (BMC) / Management Engine (ME)
  - Host / Application Processor
  - Storage / Network Controllers
  - Graphics Processing Unit (GPU)
  - Serial Peripheral Interface (SPI) Flash
  - Power Supplies
  - Etc.

Platform Firmware

- Critical to the trustworthiness of a system
- Highly privileged in system architectures
- Difficult to repair because firmware is necessary for the system to operate

The Impact

- **A successful attack on the platform firmware could**
  - Render a system inoperable, perhaps permanently
  - Inject persistent malware
  - Extract data
  - Require reprogramming by the OEM
  - Result in significant disruptions to the user

Your Brand | Your Company | Your Revenue | Your IP | Your Customers

NIST 800-193

Platform Firmware Resiliency Guidelines

- **Protection:** Mechanisms for ensuring that Platform Firmware code and critical data remain in a state of integrity and are protected from corruption, such as the process for ensuring the authenticity and integrity of firmware updates.

- **Detection:** Mechanisms for detecting when Platform Firmware code and critical data have been corrupted or otherwise changed from an authorized state.

- **Recovery:** Mechanisms for restoring Platform Firmware code and critical data to a state of integrity in the event that any such firmware code or critical data are detected to have been corrupted, or when forced to recover through an authorized mechanism. Recovery is limited to the ability to recover firmware code and critical data.

Root of Trust

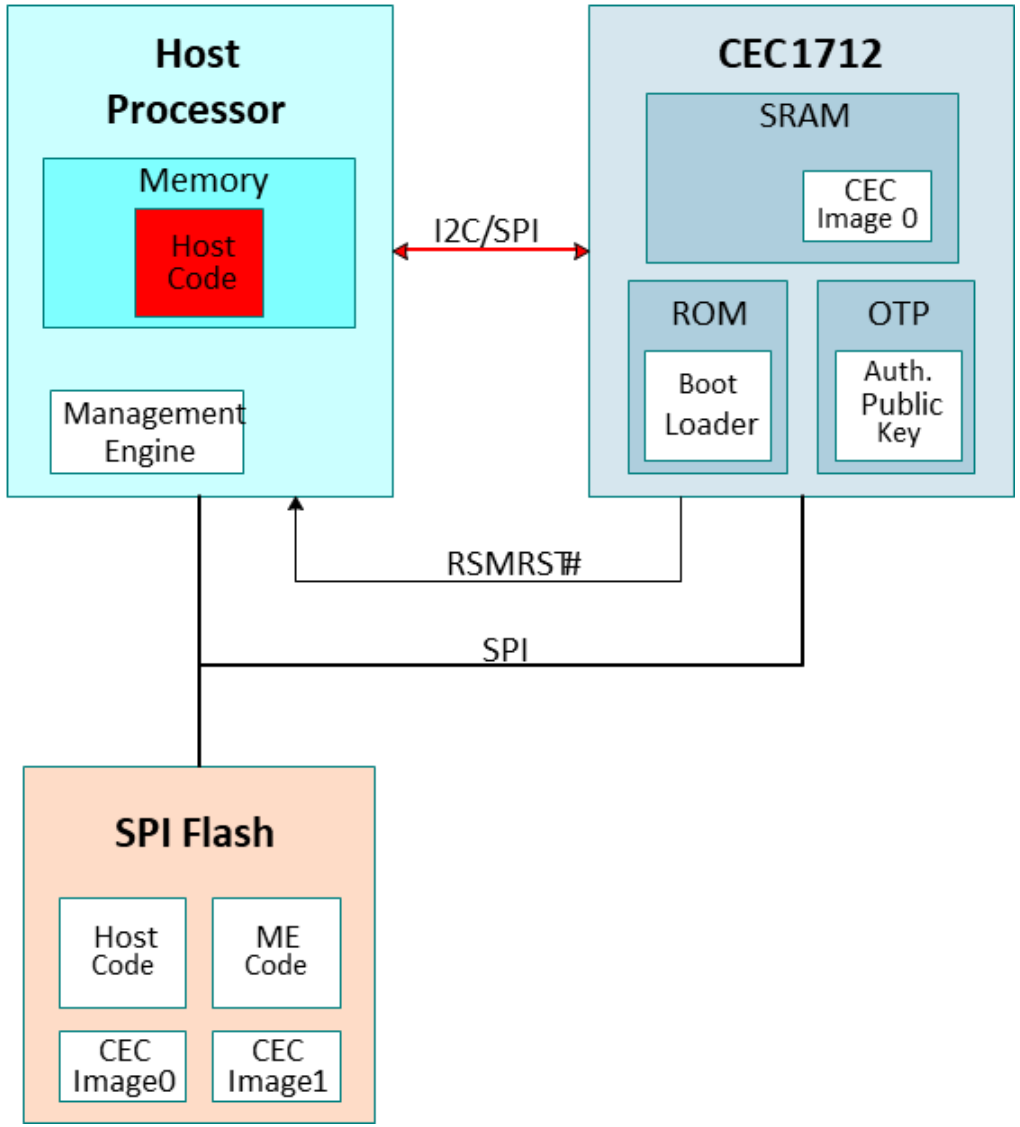
- **Platform Firmware Resiliency Guidelines based on Root of Trust**
- **Roots of Trust are secure by design:**
  - Immutable code to ensure a state of integrity
  - Digital signatures to ensure authenticity of firmware

CEC1712 and Soteria-G2 Firmware

- **CEC1712-S2/Soteria-G2 designed to enable platforms to detect and stop malicious firmware prior to run time**
- **CEC1712 hardware cryptography-enabled microcontroller**
- **Soteria-G2 firmware, when used with CEC1712, simplifies code development to speed adoption and implementation of secure boot**
- **Ensures code is authenticated BEFORE execution**
  - PREVENTS root-kit types of attacks
  - Protect against Rootkit and Bootkit malware in systems that boot from external SPI Flash Memory
- **Meets NIST SP 800-193 Guidelines**
- **Flexible and configurable to platform needs**
- **Critical for ANY design booting out of external SPI Flash**

MICROCHIP

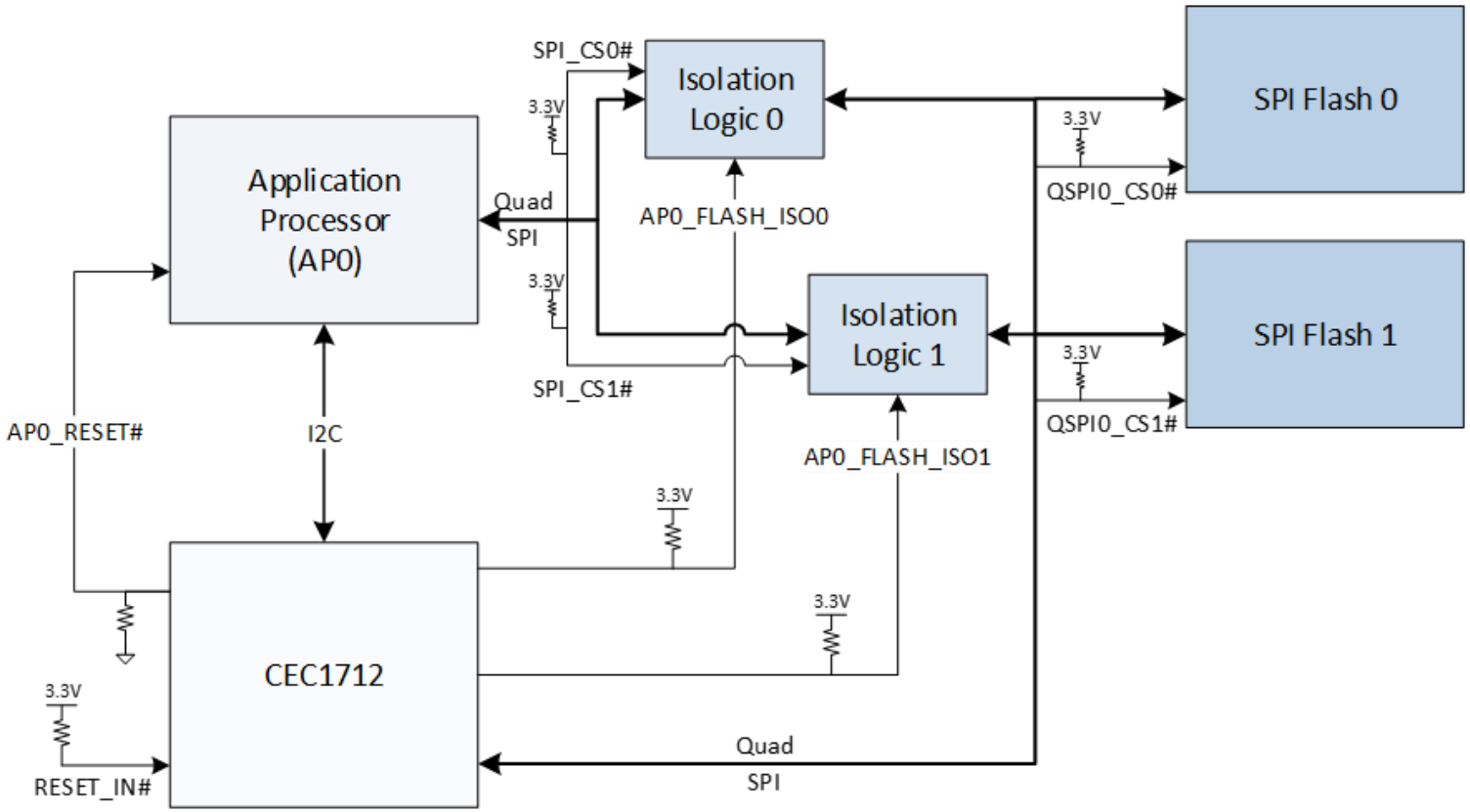
Protect / Detect



- **Code execution starts with the CEC1712 ROM**
  - Hardware based: Trusted immutable code (Boot Loader)
- **SPI Flash**
  - Code signed by OEM's private key is stored in SPI Flash (CEC Image 0 and 1)
  - Host code, signed by a private key, is also stored in the SPI Flash
- **At power-on, the Boot Loader**
  - Holds host processor in reset
  - Loads CEC1712 application code from SPI Flash into SRAM
  - Authenticates CEC1712 code using public key stored in CEC1712 OTP
- **The authenticated CEC1712 code**
  - Starts execution
  - Authenticates the host code in SPI Flash with a public key in CEC1712
  - Releases the host processor from reset
- **Host processor loads, executes its authenticated code from SPI Flash**
- **Chain of trust now extended to CEC1712 code and host code**

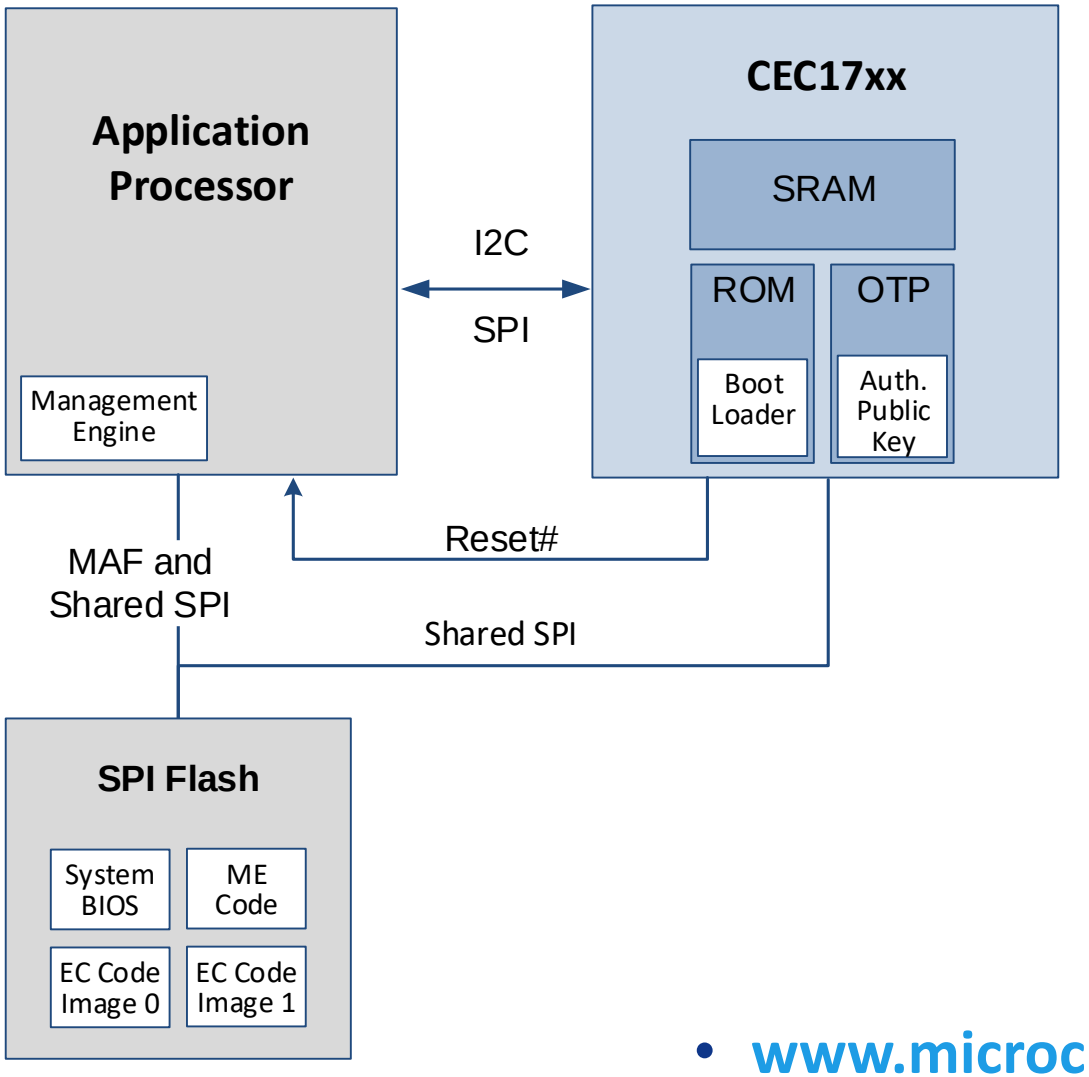
Recover

- Back-up code in SPI Flash
- With optional isolation logic, access to SPI Flash with corrupt images can be restricted



MICROCHIP

Platform Firmware Resiliency



- CEC1712 + Soteria-G2 meets NIST 800-193 guidelines: Protect, Detect, Recover
- Secure any processor using SPI Flash
- Boot ROM authenticates CEC1712 code in SPI Flash
- CEC1712 code authenticates application processor code in SPI Flash
- All authentication occurs prior to code running
- Authentication at every power cycle

- www.microchip.com/CEC1712

Thank You