# Getting Started with Secure Software

## Class 4: Secure Boot and the Root-of-Trust

April 23, 2020

Jacob Beningo

DesignNews

Presented by:

CEC CONTINUING EDUCATION CENTER

Digi-Key ELECTRONICS

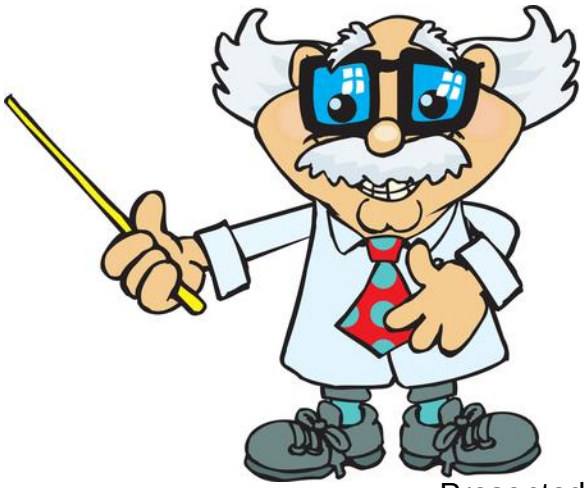
# Course Overview

**Topics:**

- Introduction to Platform Security Architecture (PSA)
- Performing a Security Threats Analysis
- Architecting a Secure Solution
- **Secure Boot and the Root-of-Trust**
- Secure Frameworks and Ecosystems

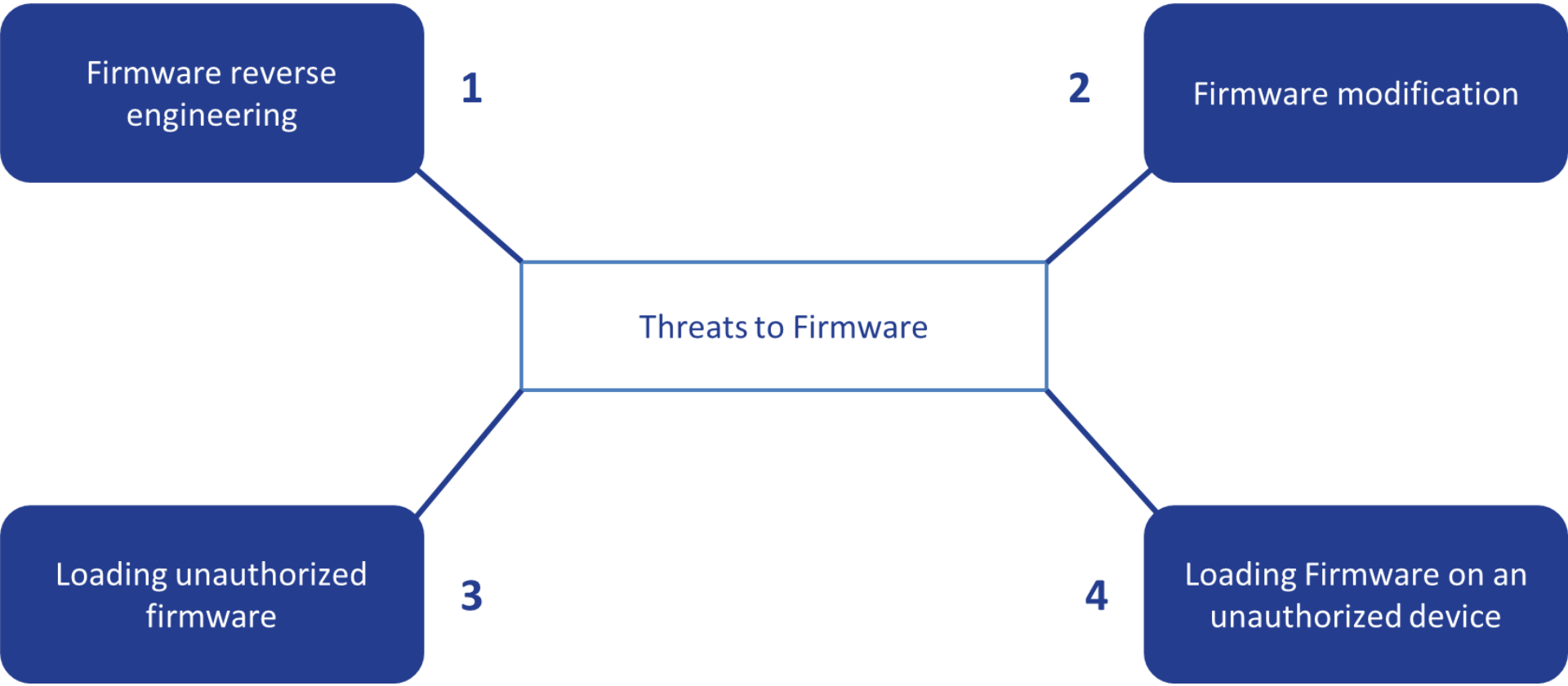
# Session Overview

- Security Use Cases
- Root-of-Trust
- Secure Boot
- Secure Bootloaders



Presented by:

# Security Use Cases



# Root-of-Trust

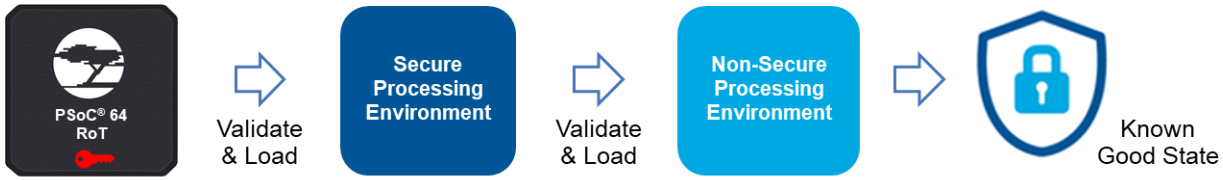
**Root-of-Trust (RoT)** – This is an immutable process or identity which is used as the first entity in a trust chain. No ancestor entity can provide a trustable attestation (in Digest or other form) for the initial code and data state of the Root of Trust.

Example:

The initial boot code stored in ROM which cannot be changed by users or Cypress provides the RoT for PSoC 64 Secure MCU's.

# PSoC® 64 Boot-time Security

## Secure Boot and Secure Firmware Updates




**Secure Boot**

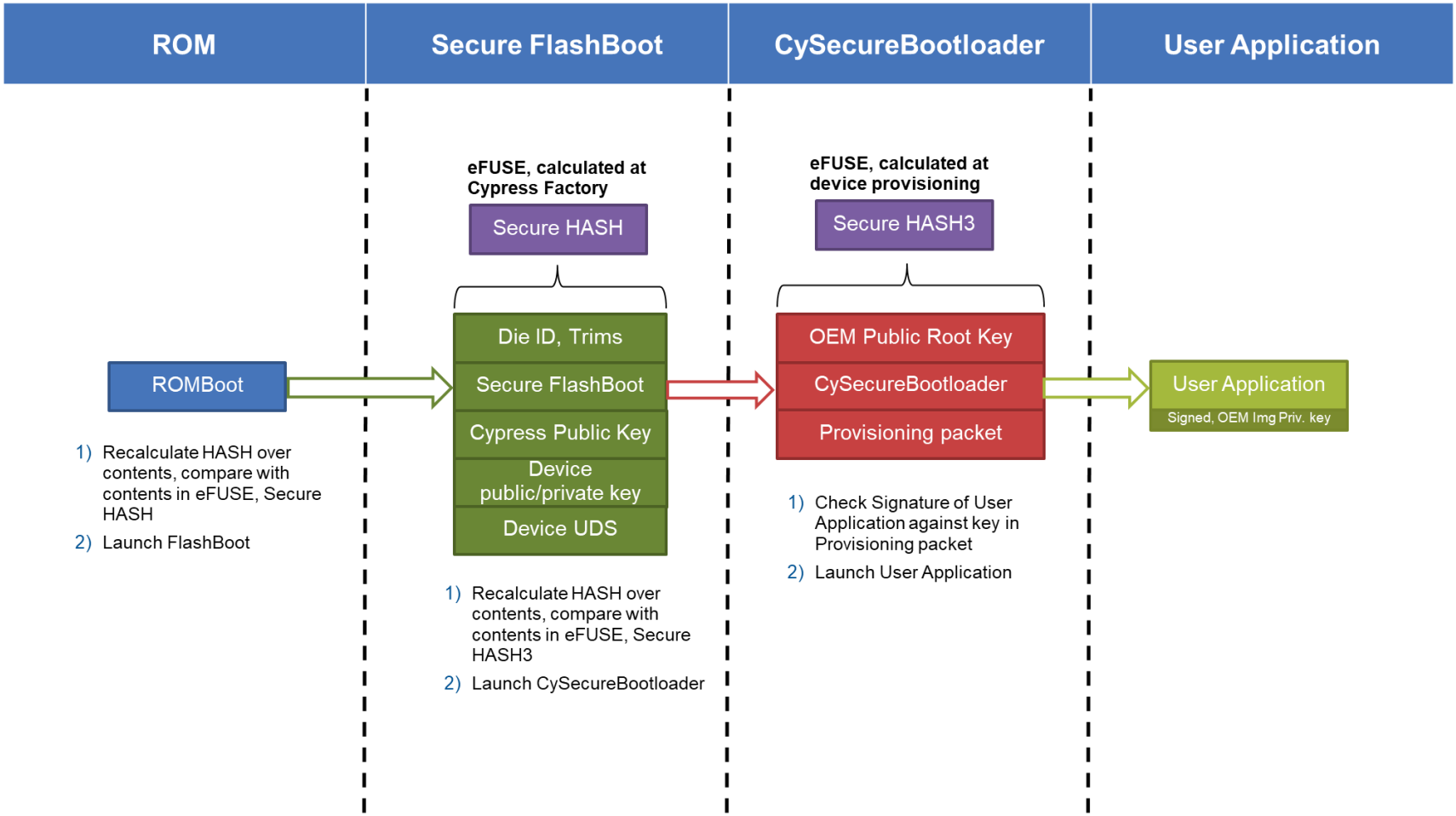
- Boot sequence validates image
  - Integrity: image has not been tampered with
  - Authenticity: image is from an authorized source
- Device boots to a known good state

**Secure Firmware Updates**

- Updated image can be stored and encrypted internally or with off-chip Quad SPI Flash
- Rollback protection prevents older firmware from being loaded with known vulnerabilities

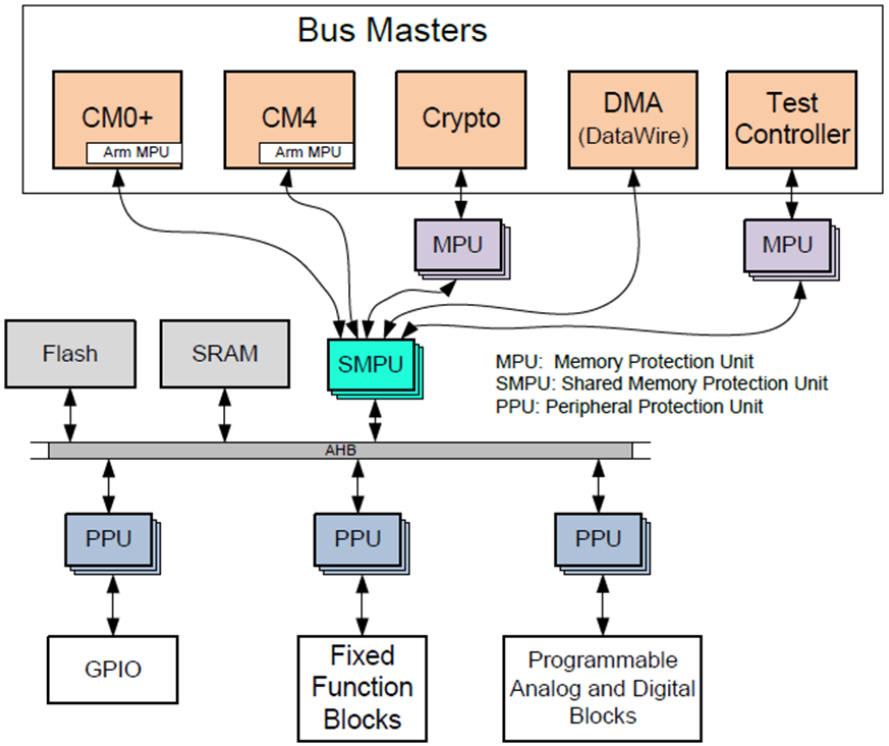
**PSoC 64 Root-of-Trust serves as the trust anchor for secure chain-of-trust**

# Secure Boot – Boot Sequence

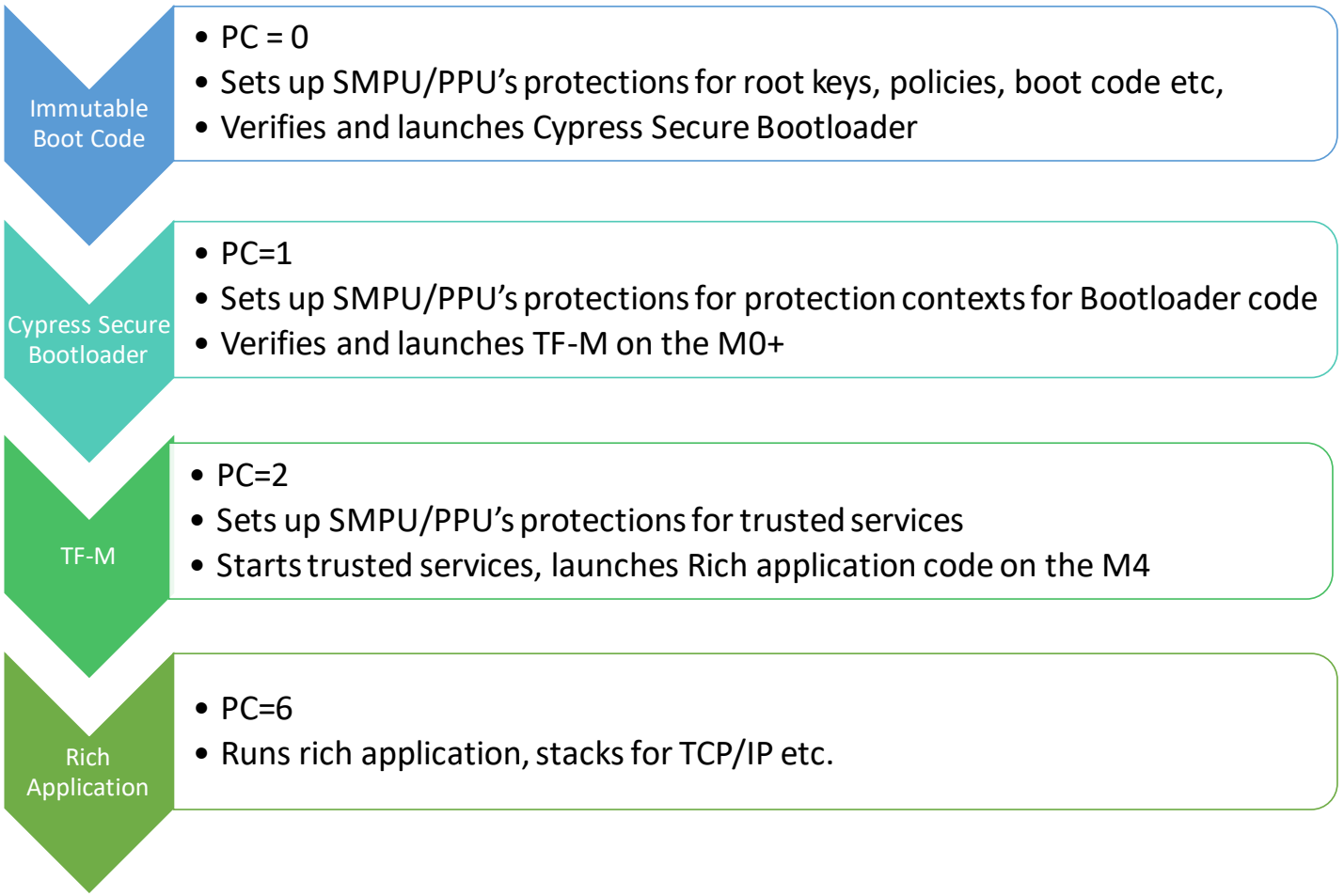


# Secure Boot – Isolation

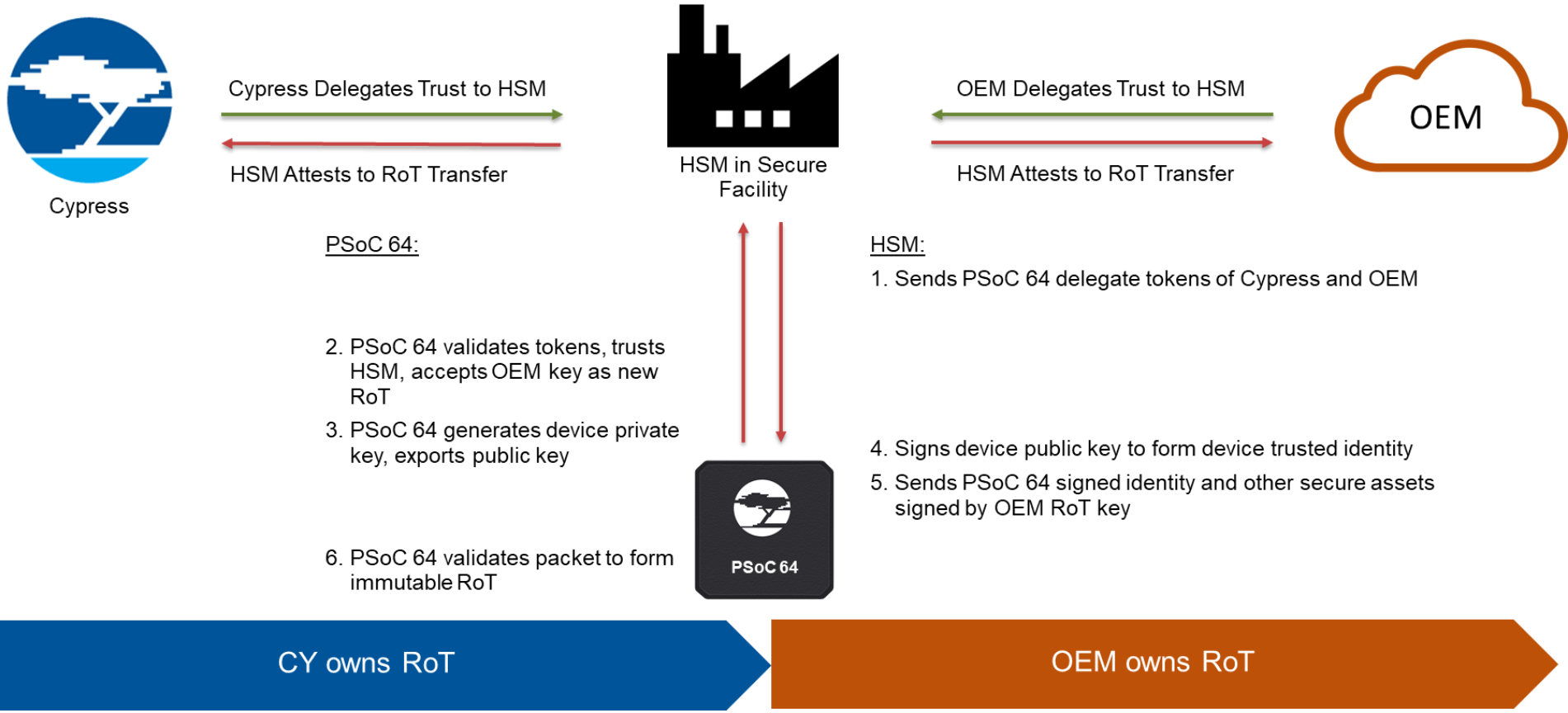
- 5 Bus Masters that can access the AHB and access Flash, SRAM and Peripherals
- Memory Protection Unit (MPU)
  - Provide high-level memory protection
  - Distinguish user and privileged access
- Shared Memory Protection Unit (SMPU)
  - Distinguishes between different protection contexts (PC)
  - Distinguishes secure from non-secure accesses
- Peripheral Protection Unit (PPU)
  - Manages access to individual peripheral blocks in different PC's, access and security states



# Secure Boot – Protection Contexts



# RoT Ownership Transfer



# Provisioning and the Chain-of-Trust

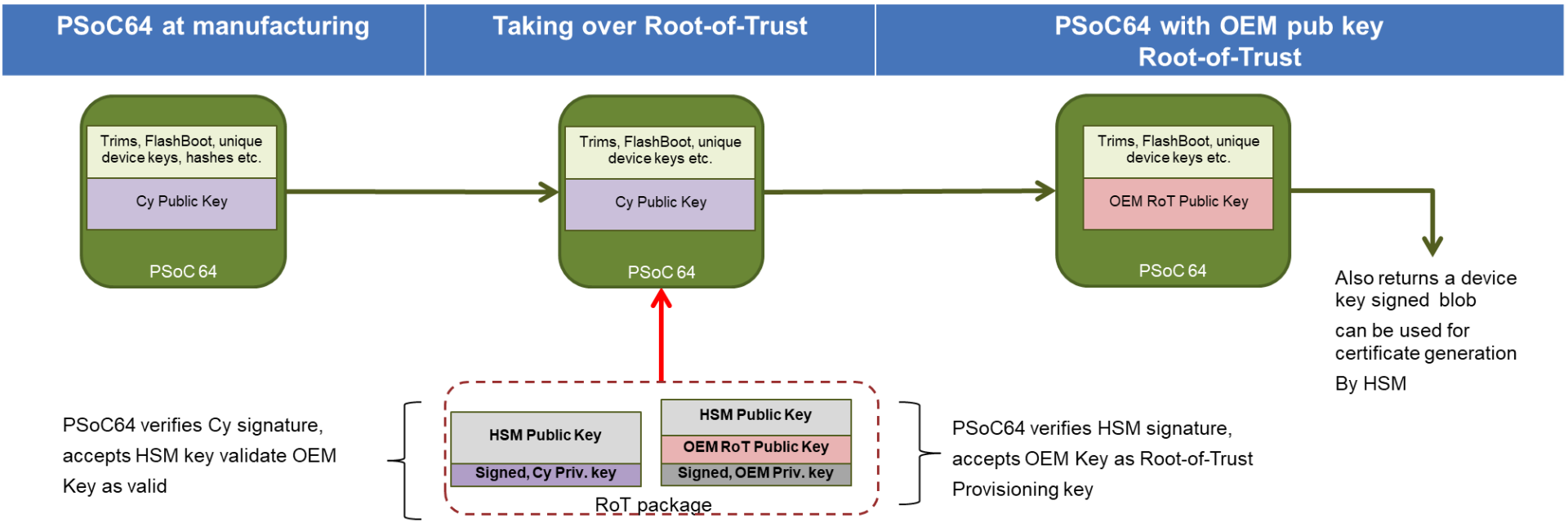
**Provisioning** - is a process where secure assets like keys and security policies are injected into the device. This step typically occurs in a secure manufacturing environment that has a Hardware Security Module (HSM). This process is irreversible. For the PSoC 64 Secure MCU, provisioning involves the following steps:

- Transferring the RoT from Cypress to the development user (called OEM in this course).
- Injecting user assets such as image-signing keys, device security policies, and certificates into the device.
- eFuses are blown (irreversible).

Provisioning the device can be done through the Cypress Secure Boot SDK.

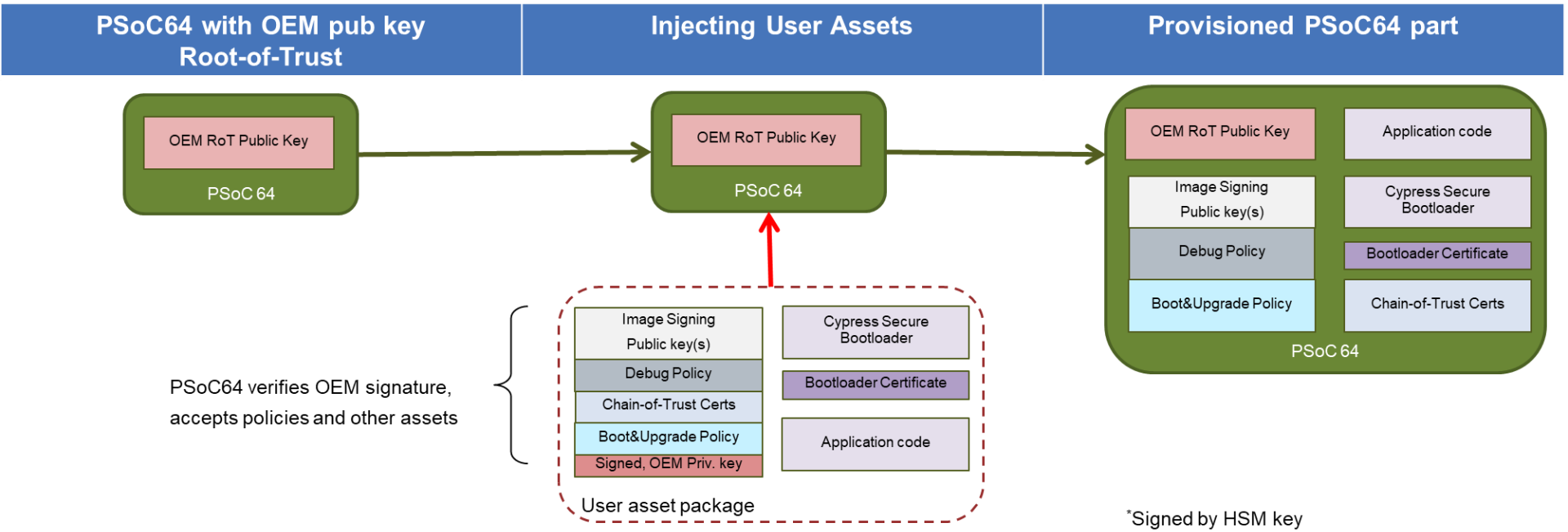
# Provisioning (Root-of-Trust)

- Every PSoC 64 has a Cypress Public key
  - Secure FlashBoot enables provisioning process
  - Provisioning requires Cypress to authorize an HSM to inject OEM key
  - HSM signs OEM public key to allow Root-of-Trust to be transferred to OEM key



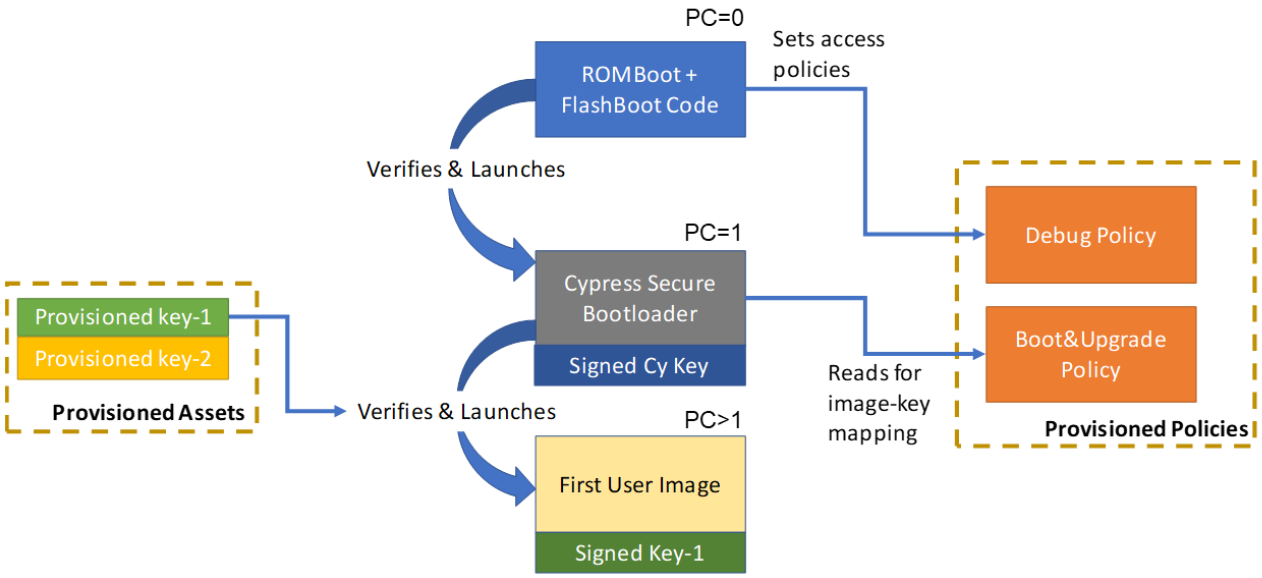
# Provisioning (User Assets)

- PSoC64 securely provisions user assets like,
  - Debug policies like, CM0+/CM4/SysAP DAP access ports
  - Image signing keys(typically are different from Root-of-Trust key)
  - Boot & Upgrade policies which specify key map to images, Slot sizes and addresses
  - Any certificates needed

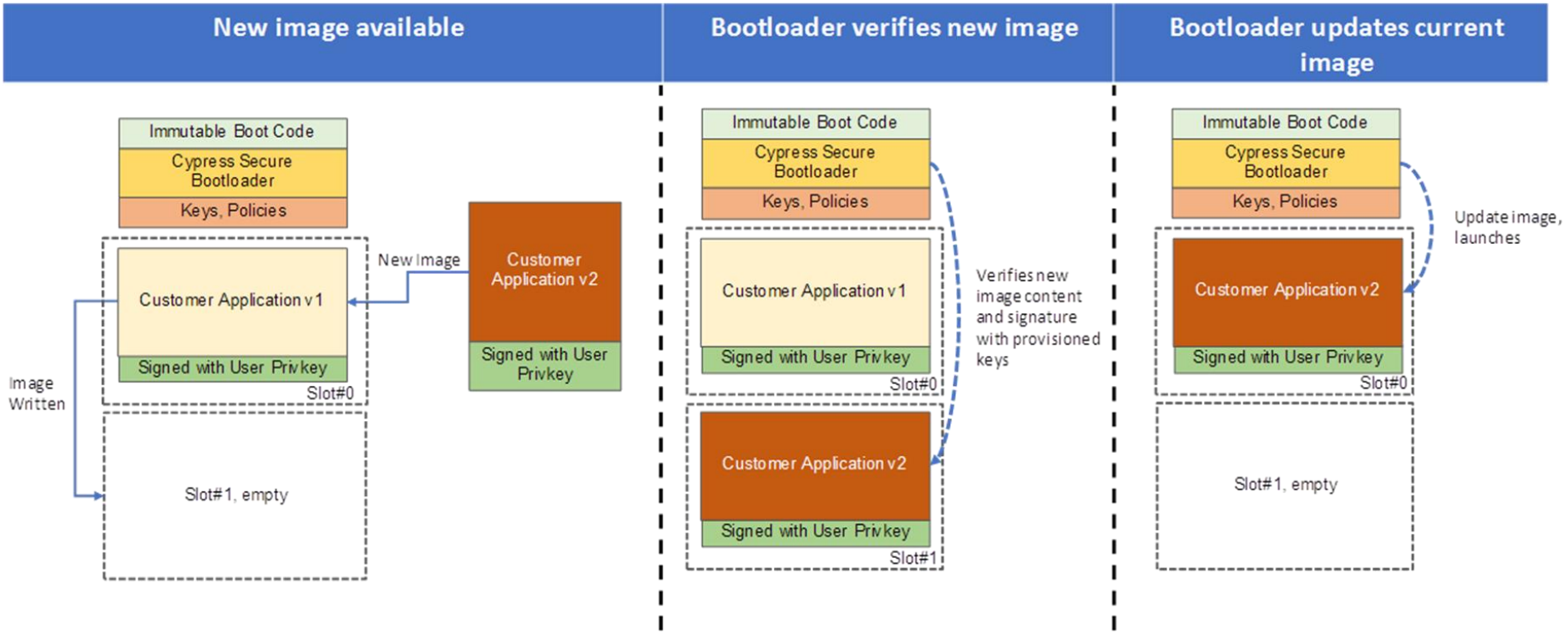


# PSoC 64 Secure Bootloader

- The Cypress Secure Bootloader is a Cypress developed piece of firmware which
  - Implements MCUBoot library
  - Has knowledge of keys & policies to setup/launch first image
  - Can be considered an extension of FlashBoot; made immutable once provisioned



# PSoC 64 Secure Bootloader



# Additional Resources

- Beningo.com
  - Blog, White Papers, Courses
  - Embedded Bytes Newsletter
    - http://bit.ly/1BAHYXm
- Platform Security Architecture:
  - www.arm.com/psa
- Threat-based analysis method:
  - www.cypress.com/psoc6security


From www.beningo.com under

- Blog > CEC – Getting Started with Secure Software