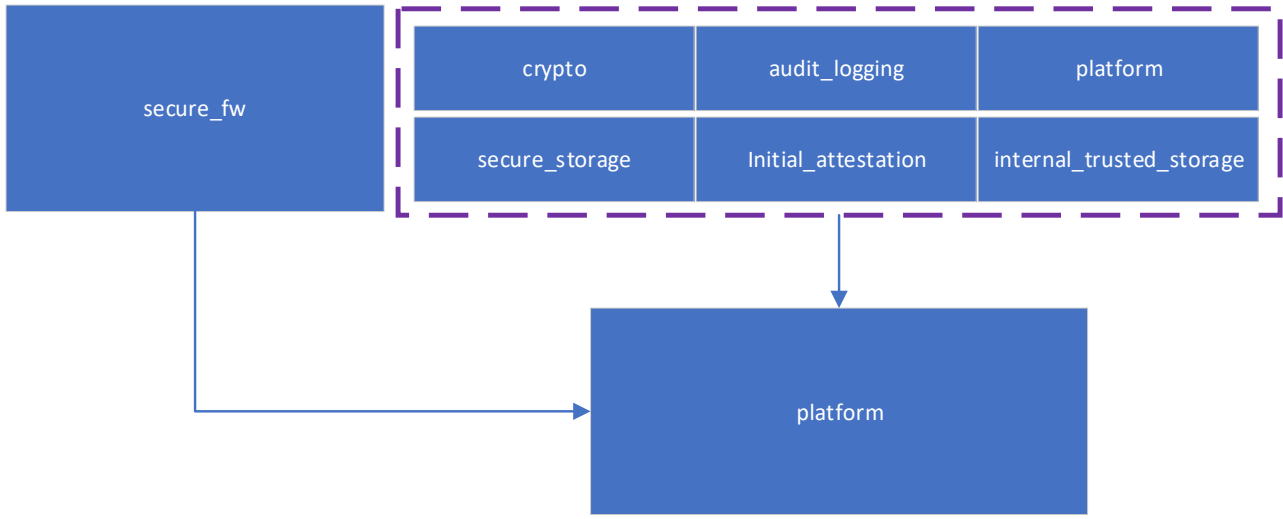
Antonio de Angelis

TF-M Hardware Abstraction Layer

TF-M platform layer

Current situation: platform directory contains all “platform related” code

- Startup files, memory layouts, driver implementation for peripherals, hardware abstraction layer (when the driver is “too complex” for a direct call) + an idea of a generic HAL interface (i.e. tfm_platform_hal_ioctl(…))
- This originated from the requirements of the secure_fw in terms of platform configuration (e.g. MPU, PPC drivers), platform testing (LEDs, timers), service requirements (e.g. read HUK, read fuses): not very organic
- Difficult to understand
  - What do I have to do to add a new platform port?
  - Which functions do I have to re-implement?
  - Which drivers are needed?



arm

TBSA-M specification

TBSA-M specification is part of the PSA set of specifications.

- A set of requirements on the platform to be deemed PSA compliant
- Public and already available on arm.com
- Covers various areas of a system (base requirements, infrastructure, boot, key handling, timers, crypto hardware, …)
- This set of requirements can be fulfilled by hardware only, by a combination of hardware plus systems software to drive the hardware, or by software only
- Testable: the psa-arch-test project contains a TBSA-M test suite: to test the requirements, the suite itself defines an hardware abstraction layer API (called the PAL – Platform Abstraction Layer)
  - It can test only those requirements which are testable through a systems software layer

arm

TF-M TBSA-M HAL

The idea is to redesign the TF-M HAL to be TBSA-M compliant

- Take into account existing platform APIs (e.g. ioctl generic interface) and ACK tests PAL layer API
- Define a new platform module in TF-M, which exposes the newly defined HAL API
- Define a new target in the TBSA-M test suite: TF-M
- All platforms which conform/expose to the new TF-M HAL API, will be testable “for free” (i.e. no additional porting required) for TBSA-M compliance (currently, one porting per target is needed, i.e. implement the PAL layer for each testable platform)
- Provides guidance for porting: A platform needs to expose to the HAL API to be able to be used on TF-M
- A self contained platform module with clear “entry points” lends itself to better “packing” of functionalities and abstraction of hardware dependencies in both secure_fw and services
- Exception: Crypto HAL needs to follow the PSA Crypto API driver model (still under development) for seamless integration with the Crypto service (mbed-crypto based)

Feedback?

Thank You!
Danke!
Merci!
谢谢!
ありがとう!
Gracias!
Kiitos!