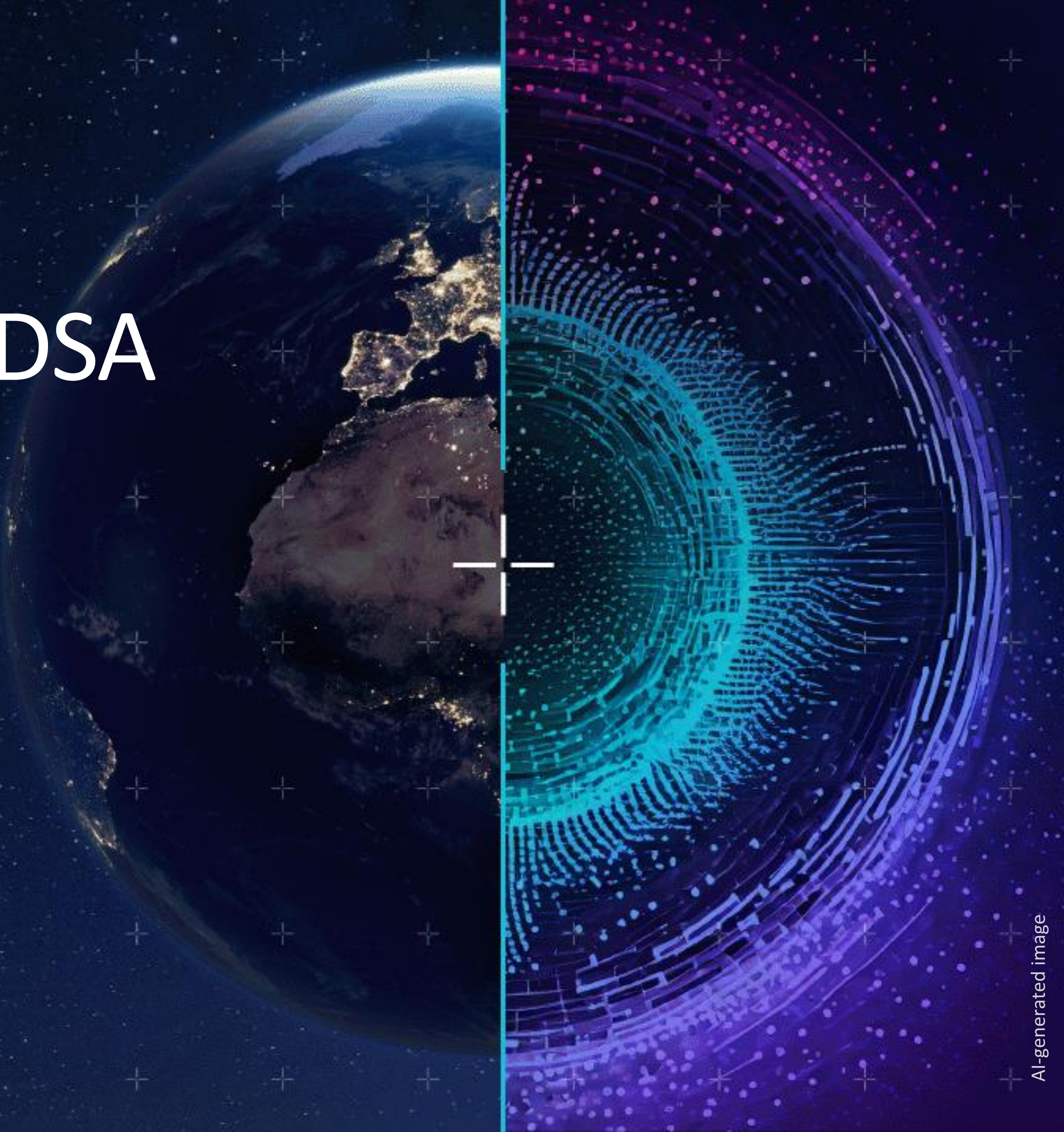
# TF-M BL2 and the ECDSA signature verification scheme

Antonio de Angelis, Arm
29/08/2024



AI-generated image

# TF-M BL2 current situation

- Based on MCUboot v2.1.0
- Enables secure boot through image signature verification
  - RSA based signature scheme available
    - 3072 bit key length (~128 bit security)
    - Security levels of less than 128 bit are not recommended for future systems
  - Based on legacy Mbed TLS APIs for crypto, i.e. MCUBOOT_USE_MBED_TLS
- HW support for Cryptographic acceleration is provided through Mbed TLS legacy APIs
  - _ALT support for link time replacement of crypto functions in the library
  - Support for _ALT functions being removed right now from Mbed TLS, i.e. next release won’t have HW support through legacy APIs

# API level change

- In the past couple of years, Arm introduced support upstream in MCUboot for MCUBOOT_USE_PSA_CRYPTO, i.e. BL2 performs cryptography through PSA Crypto APIs
- Recommended API available in Mbed TLS starting 4.0 release (2025)
- Only API that will support HW acceleration throughout PSA Crypto driver wrappers layer
  - Simplifies HW integration and management wrt legacy solution being removed
- MCUBOOT_USE_PSA_CRYPTO allows for the usage of additional mode of operations of the crypto layer, in particular usage of builtin keys (MCUBOOT_BUILTIN_KEY option)
  - See reference Arm platforms for example
- Arm has already migrated all the `platform/ext/target/arm` platforms to MCUBOOT_USE_PSA_CRYPTO

# Signature scheme change

- As part of this, we are migrating from RSA-3072 to ECDSA based on the P256 curve
  - Same security level (~128bit security)
  - Smaller keys (3K vs 256 bits)
  - Availability of hardened implementations against side channel attacks in HW
    - e.g. CC3XX driver
  - Capability to scale more effectively in the future in terms of key size for same security level
  - Availability of an efficient SW implementation of ECDSA over P256 curve, i.e. P256m
- TF-M does not mandate the usage of EC-P256 signature scheme
  - Platforms can still enable MCUBOOT_USE_PSA_CRYPTO and rely on the RSA-3072 signature scheme
- But due to API changes at previous slide, if a platform wants HW support in BL2, it must:
  - Enable MCUBOOT_USE_PSA_CRYPTO
  - Provide an HW accelerator based on the PSA Unified Driver API model
    - e.g. CC3XX driver

# Current work

- Patch on review will enforce a few items described, mainly
  - Drop ALT support in BL2
  - Enable PSA Unified Drivers in BL2 when MCUBOOT_USE_PSA_CRYPTO
- This is the first step of dropping completely support for ALT drivers in the whole TF-M project
  - Start with BL2 which has a limited number of APIs
  - If you are still using the ALT drivers, please provide support for
    - _hash_setup(), _hash_update(), _hash_finish(), _hash_abort()
    - _verify_hash()
  - Otherwise your platform will fallback to SW based crypto in BL2
  - NOTE: Support in MCUboot for MCUBOOT_ENCRYPTED_IMAGES feature is still missing. Once available, you will have to provide also HW driver implementation for _asymmetric_decrypt()
- Reminder: by the time Mbed TLS 4.0 gets integrated (sometimes in 2025), PSA Crypto + PSA Unified Drivers will be the only way to have Crypto acceleration

Thank You
Danke
Gracias
Grazie
谢谢
ありがとう
Asante
Merci
감사합니다
धन्यवाद
Kiitos
شكرًا
ধন্যবাদ
תודה
ధన్యవాదములు