
Linaro Connect BKK19-208

# Trusted Firmware M – what's cooking?

Miklos Balint
Arm

# Trusted Firmware M overview

- Open source/open governance project
- Implementation of Platform Security Architecture (PSA)
- Publicly launched at Linaro Connect HKG18
- Hot issues
  - Two flavours of SPM design
  - Dual-core isolation support
  - Interrupt handling
  - … and many more
- Aiming for wide coverage and collaborative development

# PSA security domains

**Secure domain**

Basic isolation – create a Secure Processing Environment

**Protected TCB**

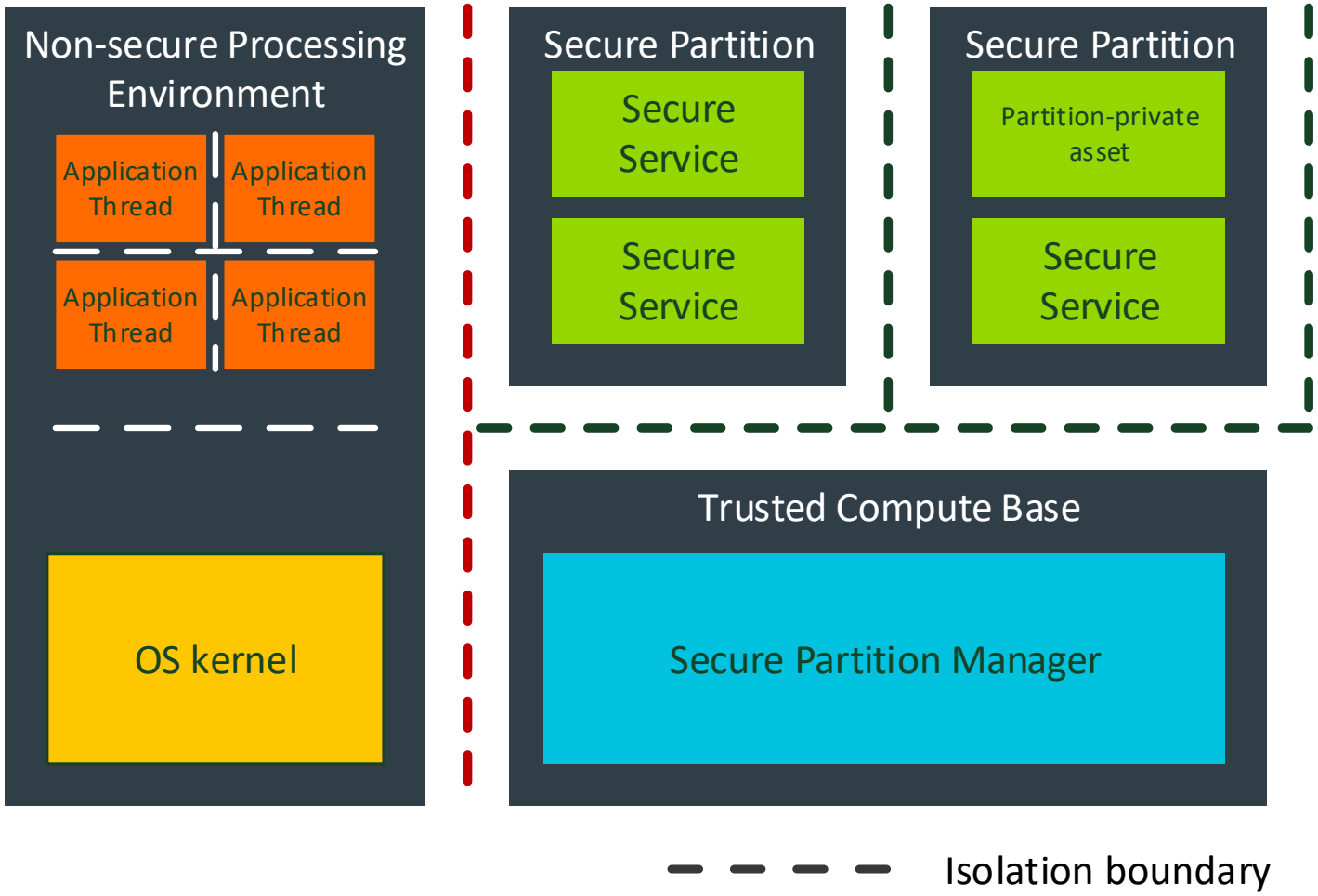
Separate Root of Trust from Secure Partitions within SPE

**Multiple tenancy in SPE**

More robustness – isolate all partitions from each other

**Non-Secure isolation**

Access policies for NS threads






arm

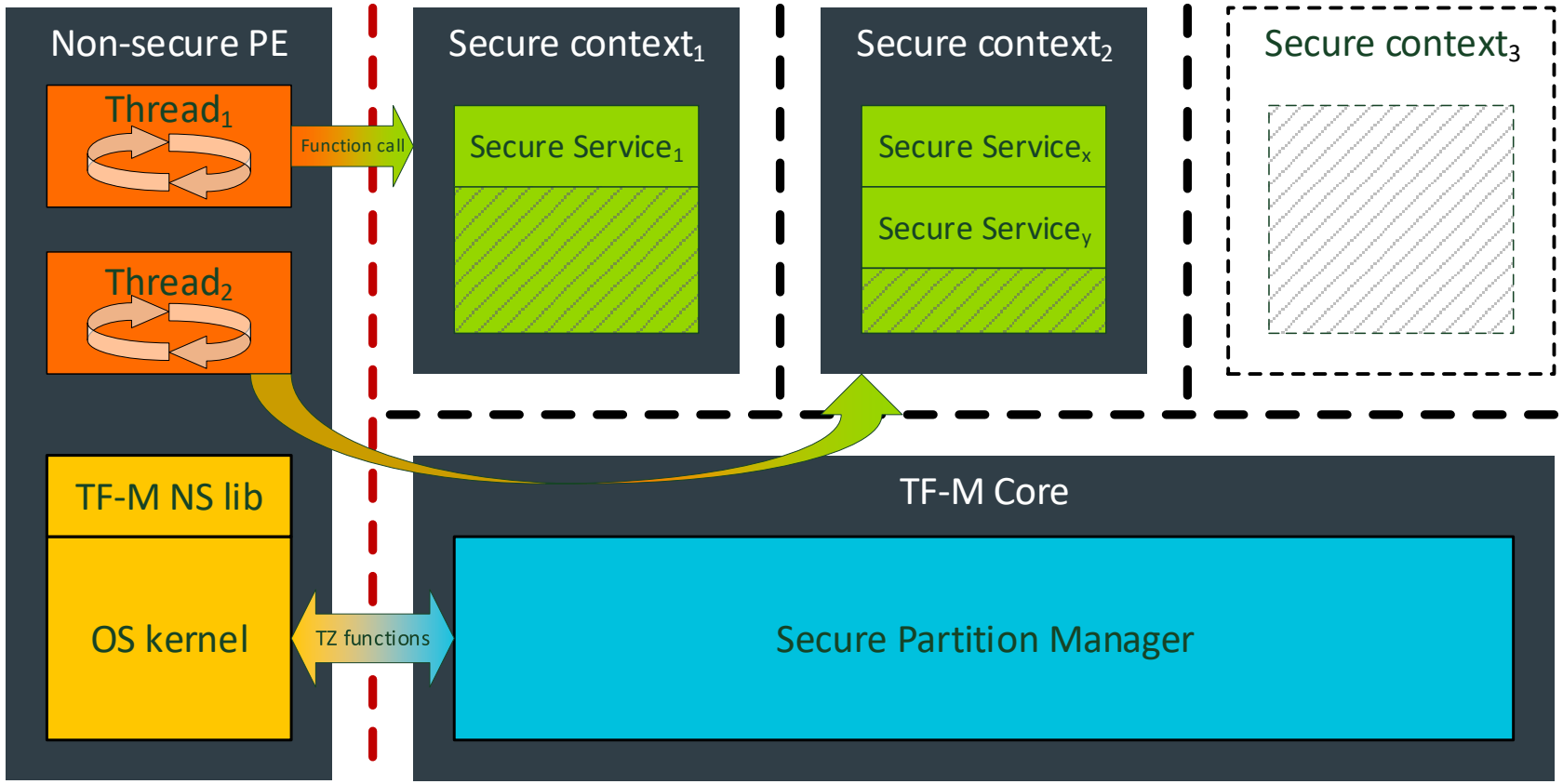
arm

# Implementations

# Trusted Firmware M shared memory model

## Secure Services implemented as functions

- Native to Armv8-M architecture
- Function call based interaction
- “Library” programming model
- Design work on…
  - Exception handling
  - Dynamic allocation of resources



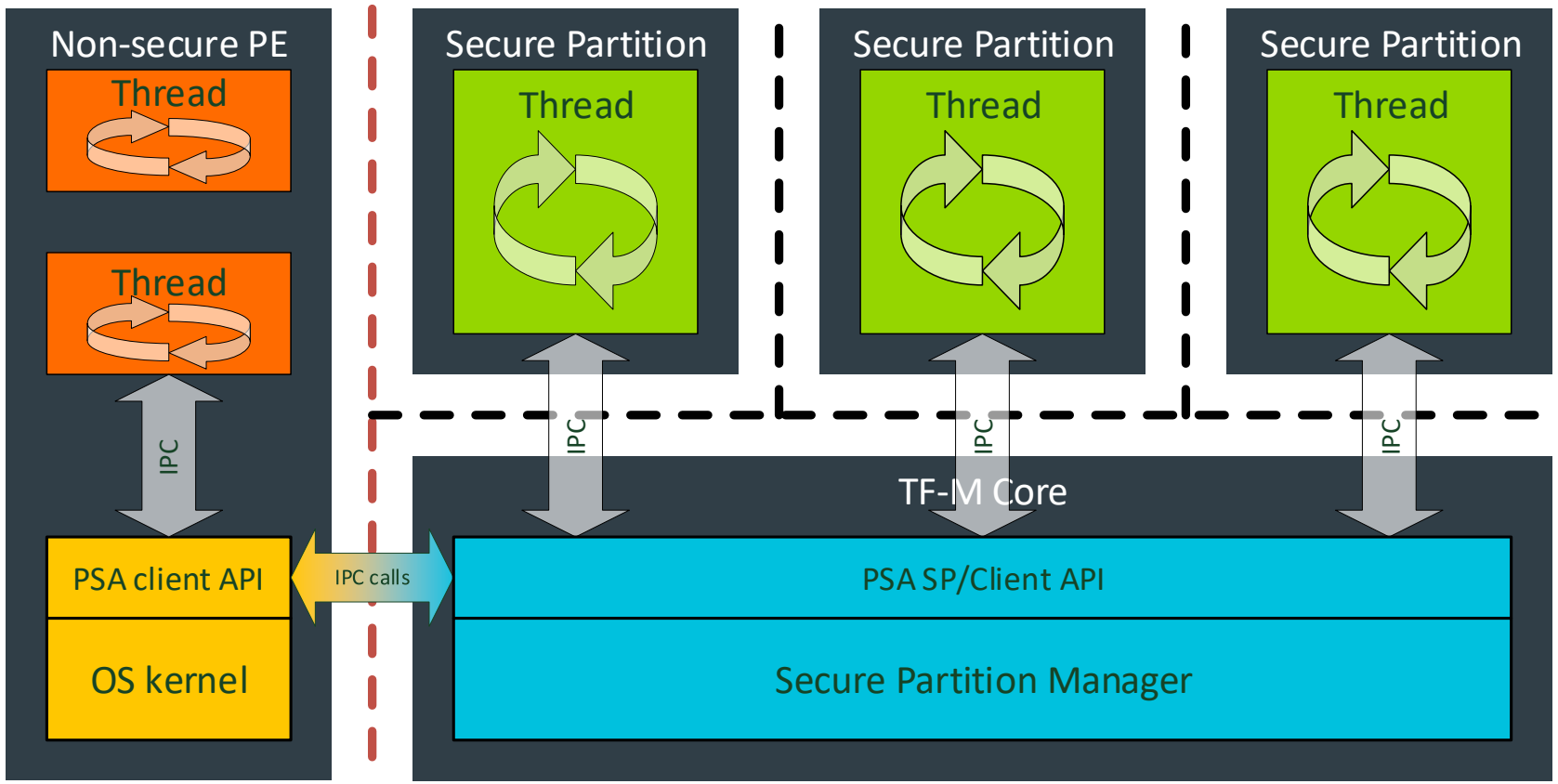



arm

# Trusted Firmware M IPC model

## Secure Partitions implemented as threads

- Robust, more prescriptive framework
- Architecture agnostic
- Static allocation of secure resources
- Connection/message based interaction
- Design work on…
  - Potential extensions



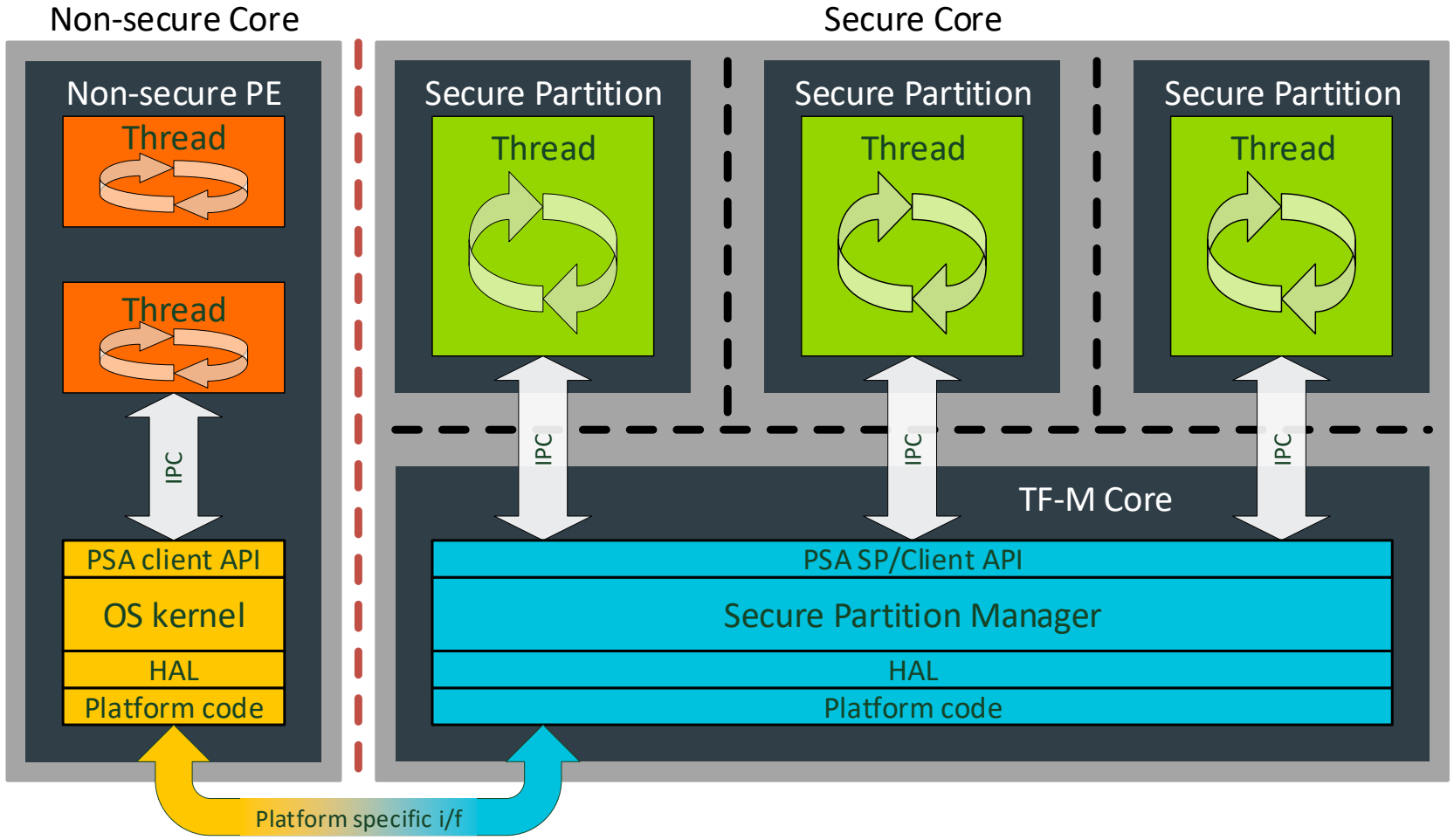



arm

# Trusted Firmware M – dual core PoC

## Physical isolation of secure and non-secure PE

- Physical isolation between secure / non-secure PEs
- Platform-specific shared resources
- Concurrent execution
- Programming model(s) same as for single core
- Design work on…
  - Interaction abstraction






arm

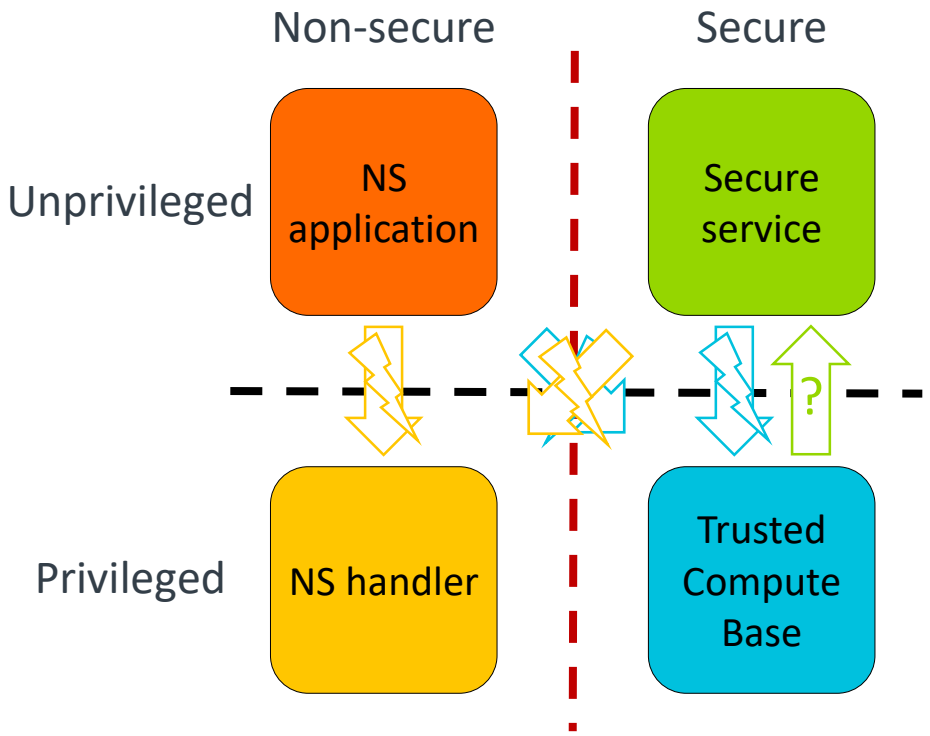
arm

# Exception Handling

in Armv8-M

# Exception flows

## and how they make life harder



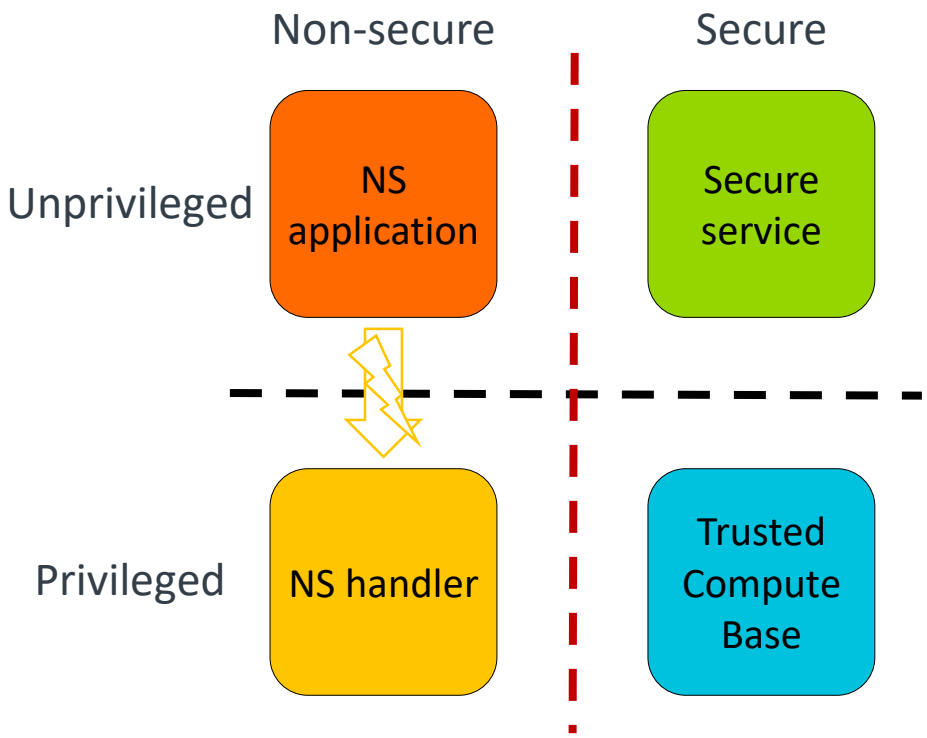



arm

# Exception flows

## Non-Secure execution interrupted by Non-Secure interrupt

- Actually that’s fine



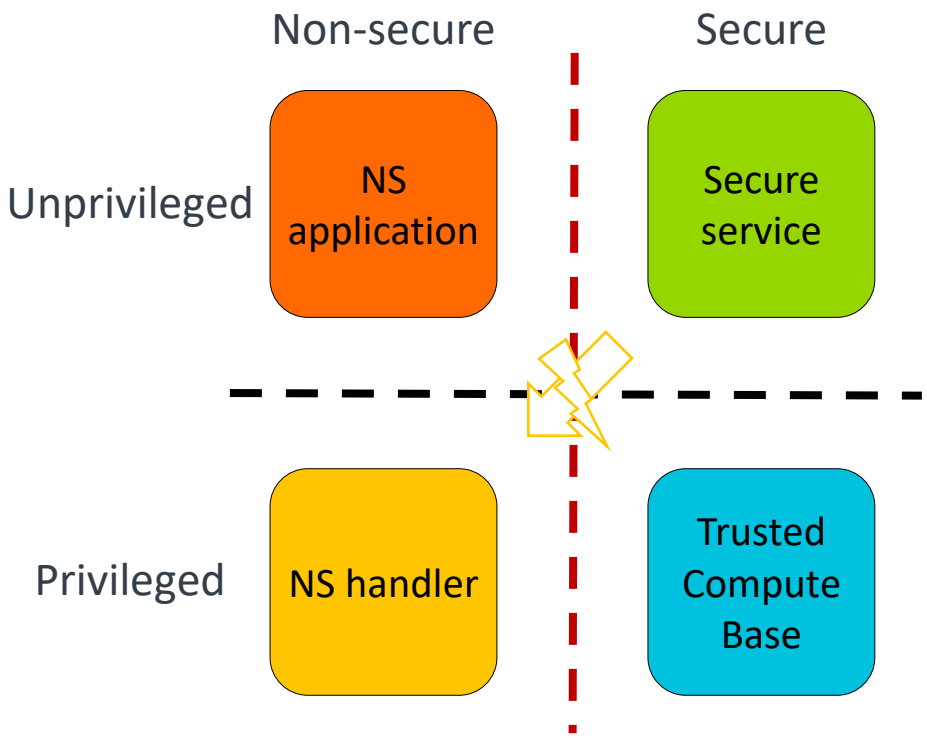



arm

# Exception flows

## Secure Service interrupted by Non-Secure interrupt

- Secure state consistency
  - Essentials provided by hardware
  - RTOS notification to TF-M about active NS ctx.
- Starvation of Secure services
  - Live with it if NS requested the service
  - Priority boost or watchdog for critical operations



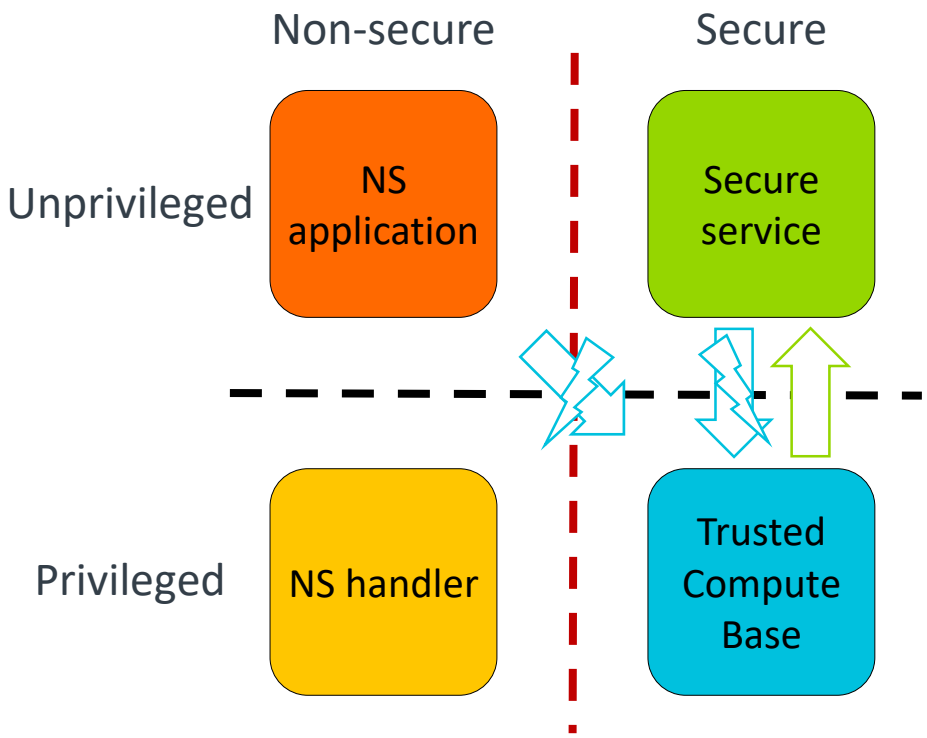



arm

# Exception flows

## Secure interrupts

- Secure Privileged interrupt handlers
  - Unrestricted access to all resources
- Principle of least privilege
  - ISR in secure container



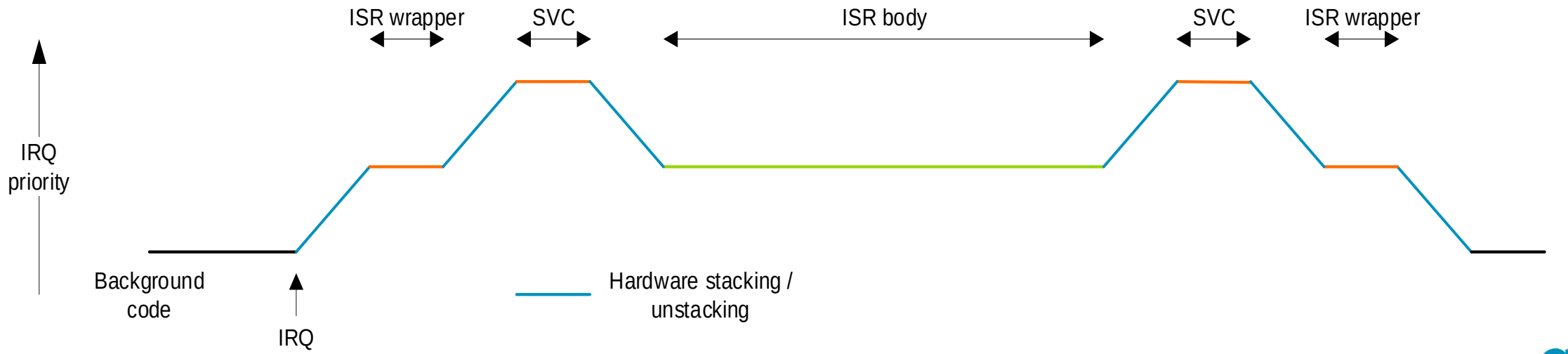



arm

# Secure interrupt deprivileging

## Device driver in secure container

- Privileged ISR owned by TF-M is wrapper
  - Triggers Partition Manager
- Sandbox created
  - Returns to thread mode
- Secure Partition code
  - Executes deprivileged ISR






arm

arm

# Summary

# TF-M design – No one-size-fits-all

## Flexible configuration is key

- Isolation
  - Physical or temporal
  - Various levels
- Execution
  - Synchronous
  - Asynchronous
  - Concurrent
- Interaction
  - Function calls
  - IPC
  - Platform-specific hardware
- Exception policies
  - … all depends on use-cases




arm

# Status

## Feedback welcome at every stage

- Shared memory model and IPC model
  - Both supported by TF-M SPM
  - Secure Service porting ongoing for IPC
- Secure interrupt handling
  - Design proposal published
  - Implementation on review
- Multiple contexts
  - NS Context awareness design and prototype published
  - Pre-emption and concurrency at design stage
- Modularization and dual core support
  - Several design documents on review
  - PoC work ongoing
  - Stay tuned for next talk

# How to get involved

- TF-A and TF-M master codebases
  - https://git.trustedfirmware.org/
- Arm contacts @ Connect BKK19
  - Abhishek Pandit
  - Shebu Varghese Kuriakose
  - David Wang
  - Karl Zhang
  - Miklos Balint
- Get in touch
  - Come round LITE hacking room between 2-3 pm on Thursday
  - Schedule a meeting
  - Contact TF-M team at support-trustedfirmware@arm.com
  - Subscribe to the TF-M mailing list at https://lists.trustedfirmware.org/mailman/listinfo/tf-m

More info at developer.trustedfirmware.org



arm


arm
Q&A

Thank You
Danke
Merci
谢谢
ありがとう
Gracias
Kiitos
감사합니다
धन्यवाद
شكرًا
תודה