Linaro Connect
MADRID 2024 | MAY 14-17 2024

Enabling the FF-A Software Standard for KVM Virtual Machines

Lorenzo Pieralisi - 16/5/2024

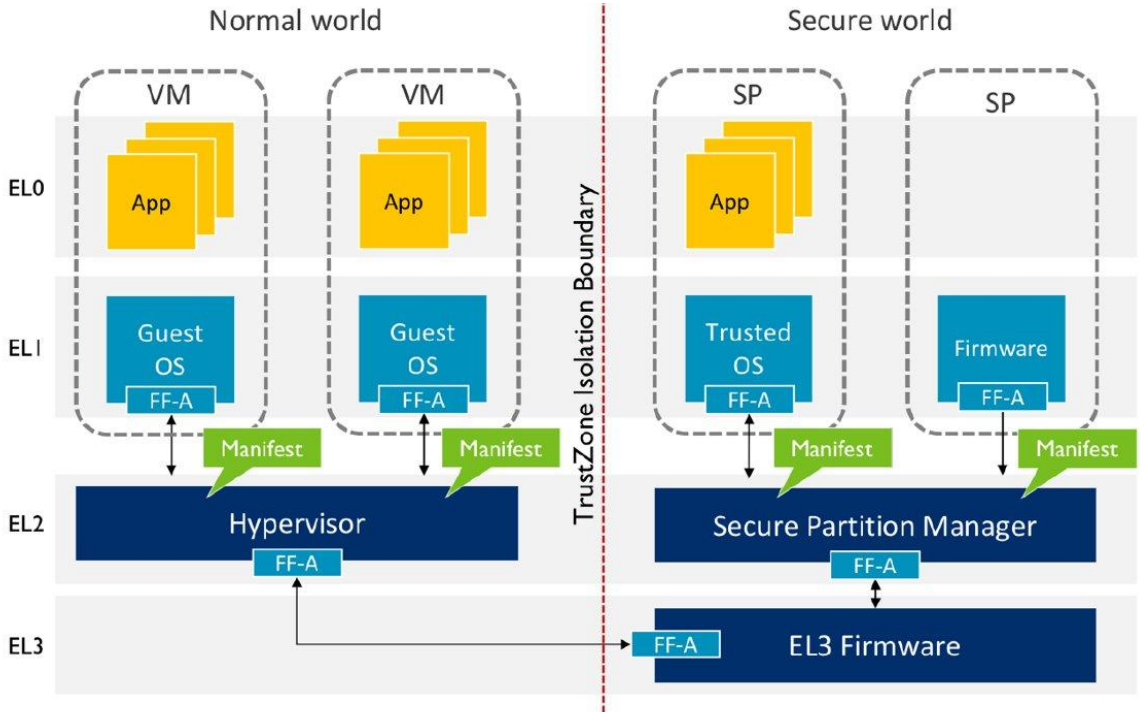
FF-A: Overview

- Firmware Framework for A-Profile processors
- Standardize communication between software images (in Normal and Secure world)
- A ***partition*** (or **endpoint**) is defined as a software module or image that implements one or more services within an isolation boundary such that a service is accessible across the boundary only via well defined interfaces
- A ***partition manager*** is responsible for creating and managing the physical isolation boundary of a partition

FF-A provides mechanism for:

- Discovering the presence of a partition, its properties and services it implements
- Synchronous and asynchronous message passing between partitions
- Memory management between partitions

Firmware Framework for Armv8-A



- One or more (secure) partitions
- Partitions manifests
- Partition managers

FF-A: Memory management

- The Firmware Framework describes mechanisms and interfaces that enable FF-A components to manage access and ownership of memory regions in the physical address space
- FF-A components can use a combination of Framework and Partition messages to manage memory regions in the following ways:
  - The Owner of a memory region can transfer its ownership to another FF-A endpoint
  - The Owner of a memory region can transfer its access to one or more FF-A endpoints
  - The Owner of a memory region can share access to it with one or more FF-A endpoints
  - The Owner can reclaim access to a memory region after the FF-A endpoints that were granted access to that memory region have relinquished their access

FF-A: Messaging

- The synchronous message passing method specified by the Framework is called ***Direct messaging***. In this method, the Sender relinquishes control to the Receiver at the time of message transmission and blocks until its receives a response from the Receiver
- The asynchronous message passing method specified by the Framework is called ***Indirect messaging***. In this method, the Sender does not relinquish control to the Receiver at the time of message transmission

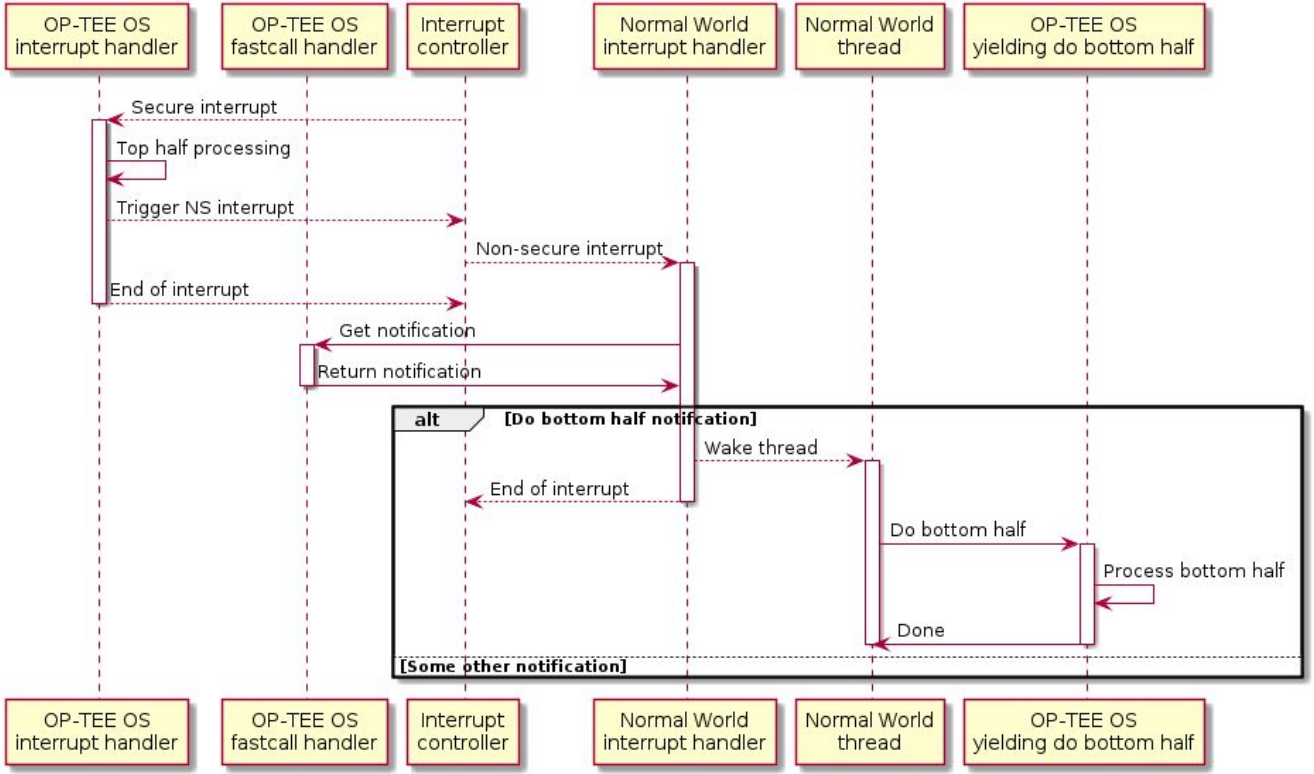
FF-A: Notifications

- The notification mechanism enables a requester endpoint (ie the Sender) to notify a service provider endpoint (ie the Receiver) about an event with non-blocking semantics
- Work underway to enable FF-A notifications (for host and VMs)
  - ***pKVM use case*** for VMs
  - Secure to Normal world notifications (Trusty IPC - OP-TEE)
  - Basis for Asynchronous messages (aka indirect messages)
- FF-A Notifications are interrupts/doorbells at their core
  - VMs discovery/probe
  - Need to inject IRQs into VMs
  - PartitionIDs vs VMIDs
  - Secure SGIs donation (aka which ones to choose)
- Ongoing work upstream to enable FF-A notifications for host and VMs
  - ***Linaro/ARM/Google collaboration***

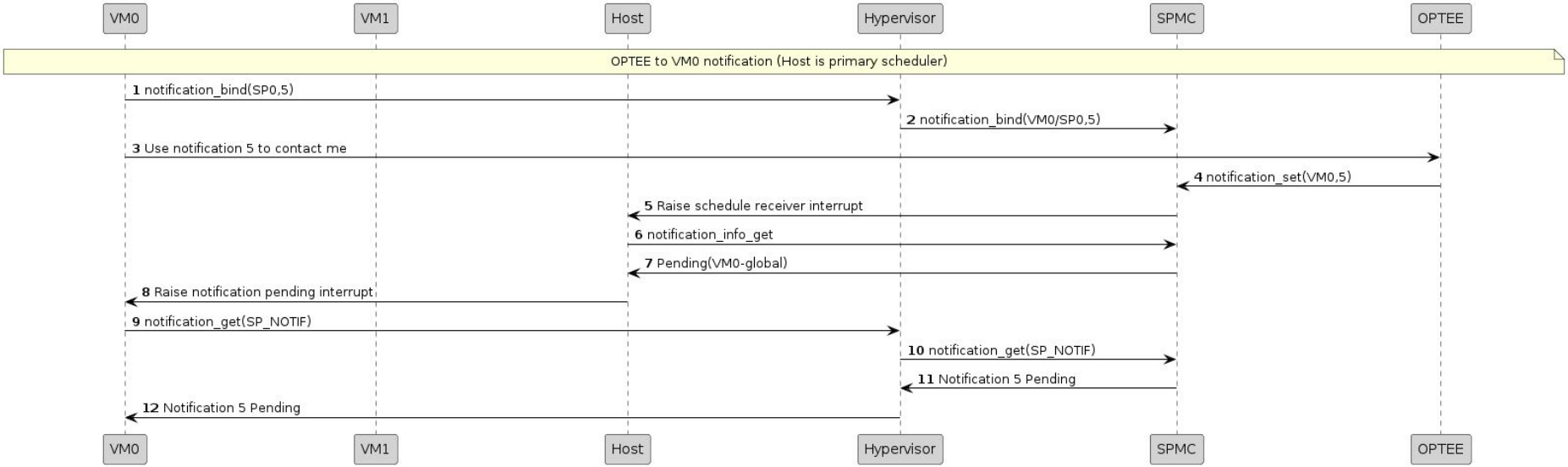
FF-A: KVM Hypervisor’s role

- Hypervisor traps guest FF-A calls and act as a relayer to secure world
- Partition ID allocation to VMs
- Memory management
  - Sharing
  - IPA repainting (IPA<->PA records)
  - Memory handles retrieval for memory reclaim
- FFA_VERSION coordination with the host FF-A driver
- TX/RX buffers mapping and handling
- FF-A notifications handled in the host (that owns the interrupt controller)
  - FFA_NOTIFICATION_INFO_GET handled in the host
  - Notification pending interrupt injected into the guest

Real world notification mechanism requirement: OP-TEE bottom-half/top-half design



FF-A: Linux VM<->SP notifications flow



FF-A handling in a Linux (p)VM: Summary

- Google patches (public but not upstream - upstreaming subject to pKVM design choices) to enable pKVM hypervisor pVM guest FF-A SMCs handling
  - FFA_PARTITION_INFO_GET
  - FFA_VERSION
  - FFA_MEM_SHARE/FFA_MEM_RECLAIM handling
  - FF-A notification calls
- OP-TEE bottom-half design working in a pVM (with FF-A ops for memory sharing and notifications set-up)
- FF-A notification pending interrupt injected from the host kernel
- Injection should move to VMM code
  - `eventfd` mechanism to relay event to VMM in userspace
  - KVM_IRQ_LINE ioctl to inject the IRQ
- OP-TEE does not support FFA_MEM_RETRIEVE_REQ
  - pKVM requires it to handle FFA_MEM_RECLAIM
- Demo on Friday !

Linaro Connect
MADRID 2024 | MAY 12-17 2024

Thank you