Architecting TEEs with RV-ACRN Hypervisor on RISC-V Platforms

Haicheng Li

haicheng.li@intel.com

intel®

Legal Notices and Disclaimers

Statements in this document that refer to future plans or expectations are forward-looking statements. These statements are based on current expectations and involve many risks and uncertainties that could cause actual results to differ materially from those expressed or implied in such statements. For more information on the factors that could cause actual results to differ materially, see our most recent earnings release and SEC filings at www.intc.com.

All product plans and roadmaps are subject to change without notice. Any forecasts of goods and services needed for Intel's operations are provided for discussion purposes only. Intel will have no liability to make any purchase in connection with forecasts published in this document. Code names are often used by Intel to identify products, technologies, or services that are in development and usage may change over time. No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

 This document contains information on products and/or processes in development.

RV-ACRN Project Info

RV-ACRN: the Port of ACRN on RV64 Architecture

https://github.com/intel/acrn-riscv

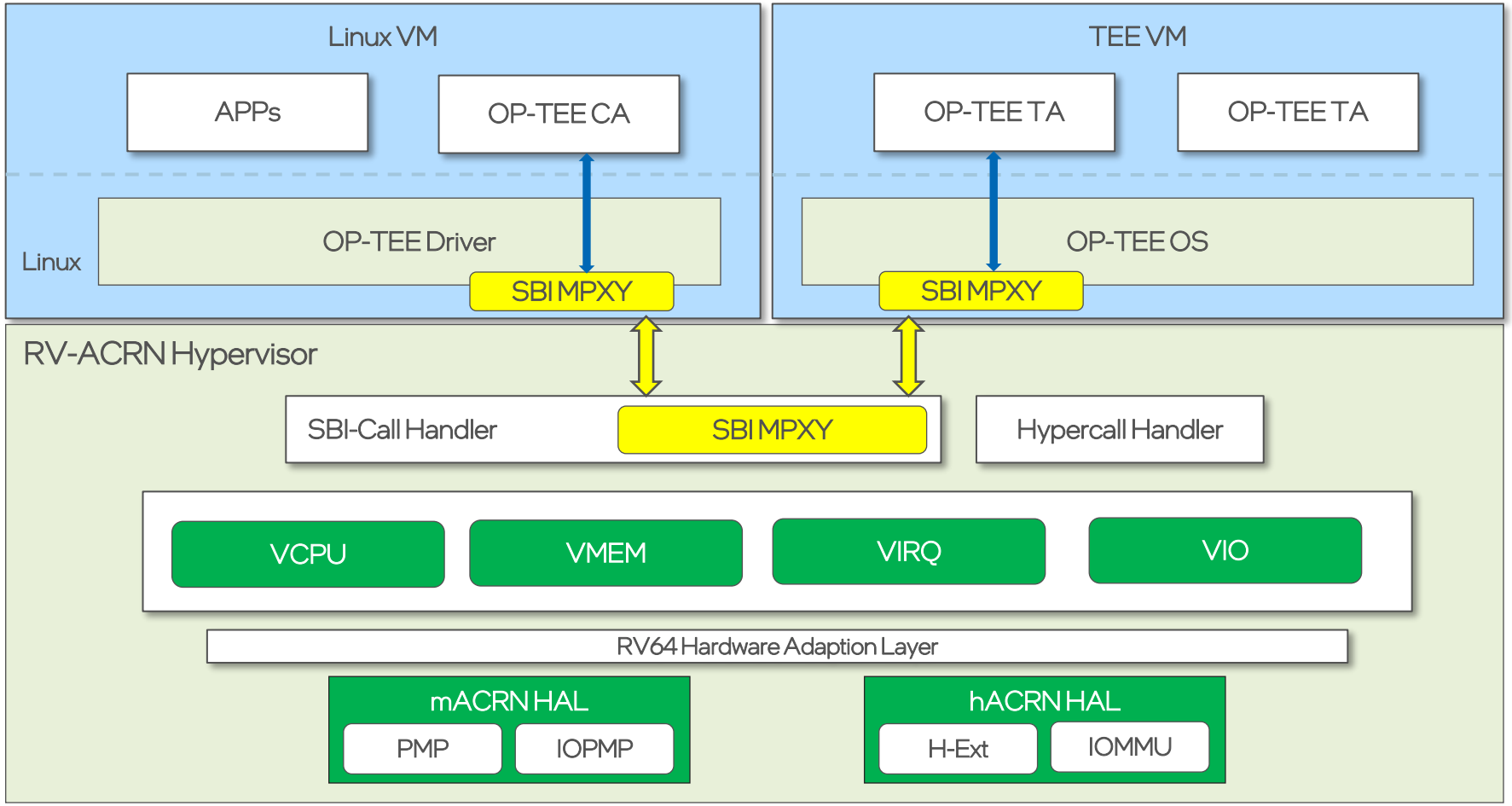
Development Branches:

- ***main****: m-mode hypervisor, Android + Linux SMP VMs run on Sophgo SG2042 machine; h-mode hypervisor, 2 VMs run on QEMU*
- ***uefi-dev****: EDK2 EFI BIOS + Ubuntu SMP VM run on QEMU*
- ***tee-dev: RISC-V TEE VM enabling***

- ACRN™ is a type-1 opensource hypervisor that runs on bare-metal hardware featured by safety, real-time and security with real production applications on x86 (IEC-61508) . https://projectacrn.org

- **RV-ACRN** is the ACRN's RISC-V port that can well run on Multi-Segment RISC-V SoCs with both **m-mode hypervisor** and **h-mode hypervisor** supported.

RV-ACRN TEE Architecture

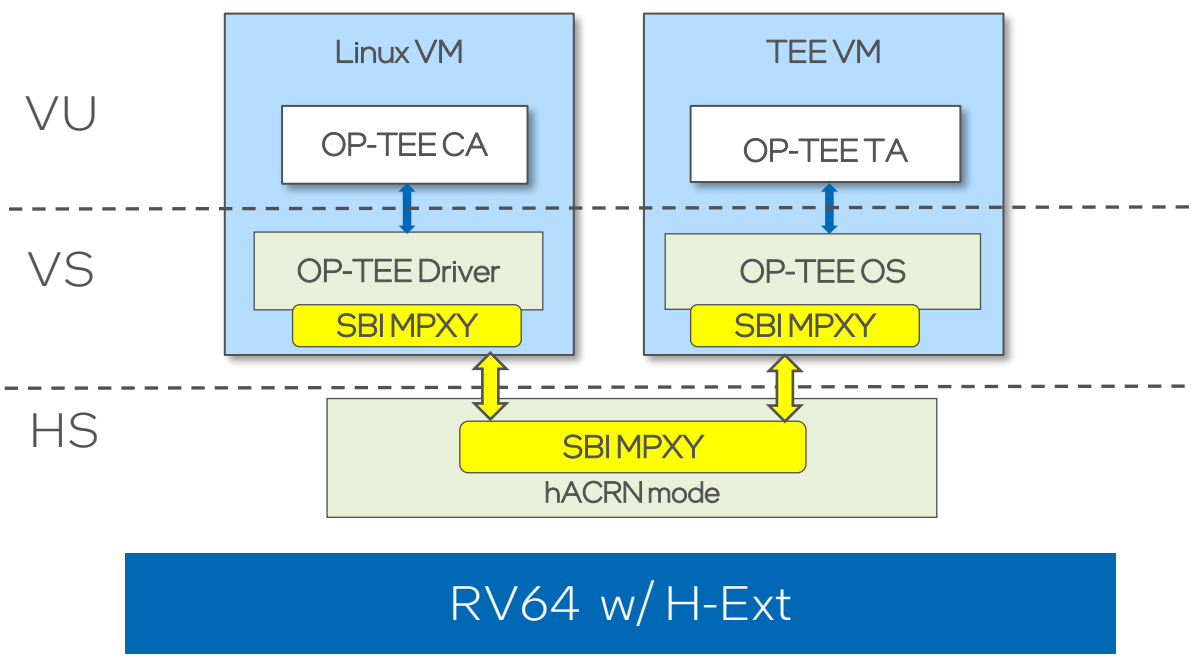


- **RV-ACRN**
  - Type-1 Hypervisor
  - SBI MPXY support
- **Linux VM**
  - Unmodified Linux
  - OP-TEE SBI MPXY Driver
- **TEE VM**
  - OP-TEE OS
  - SBI MPXY Driver

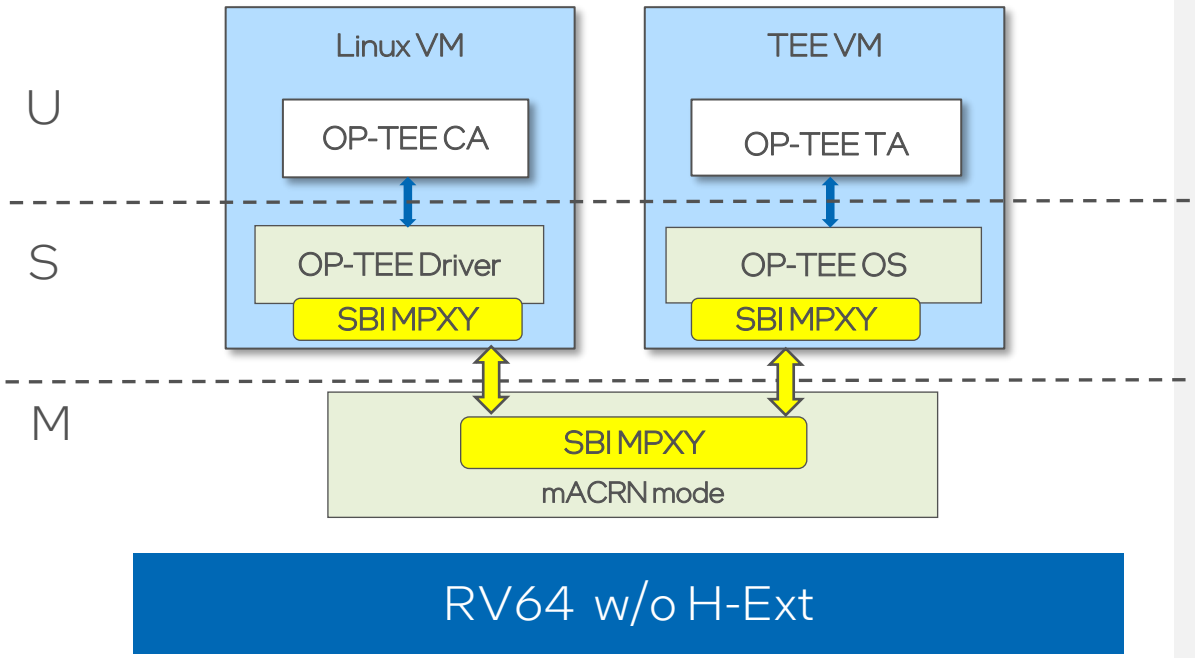
SBI MPXY Adaption

RV-ACRN Base Infra.

RV-ACRN TEE Deployment Model

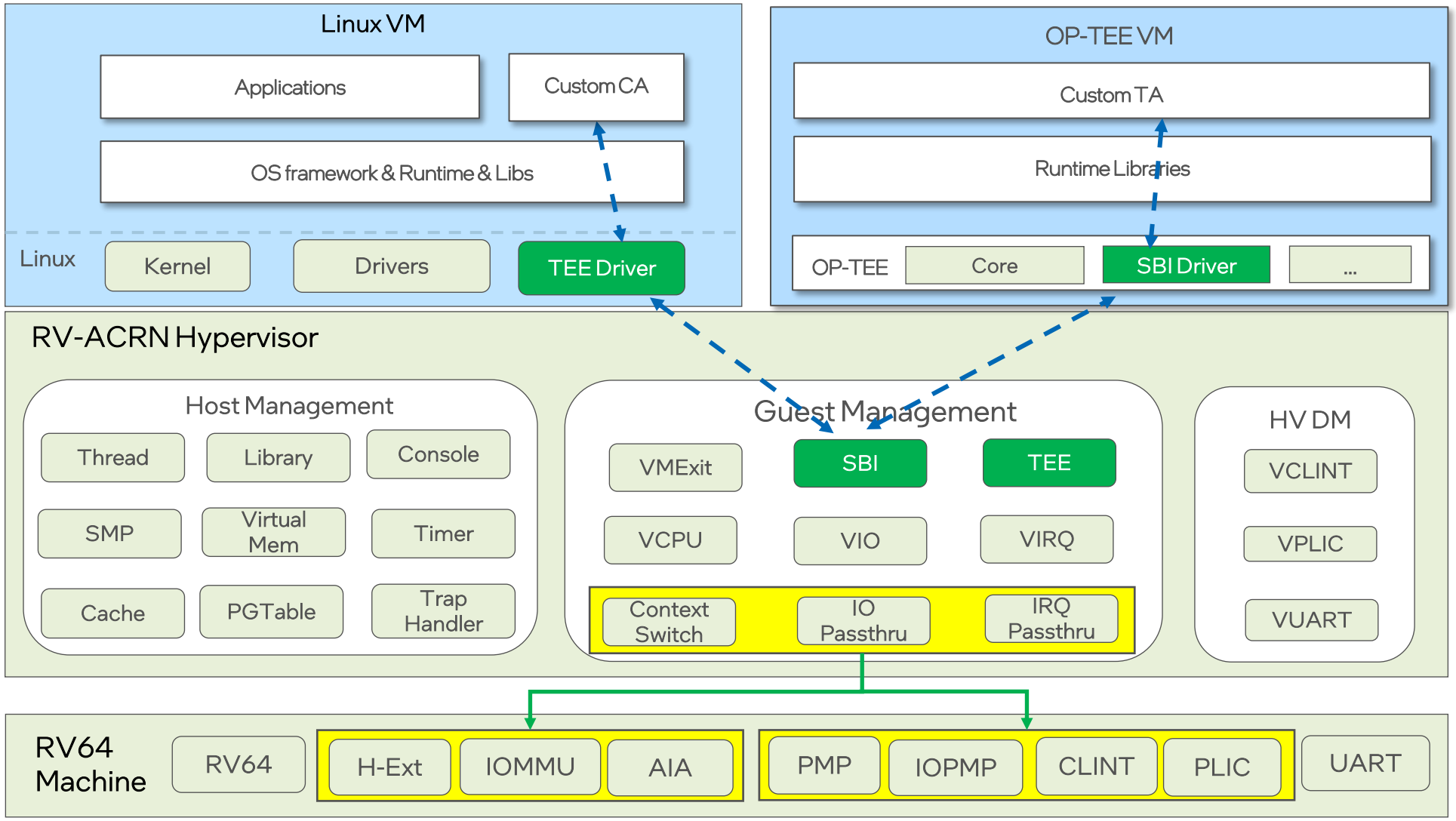


Solution1: TEE on RISC-V (H-Ext + IOMMU)



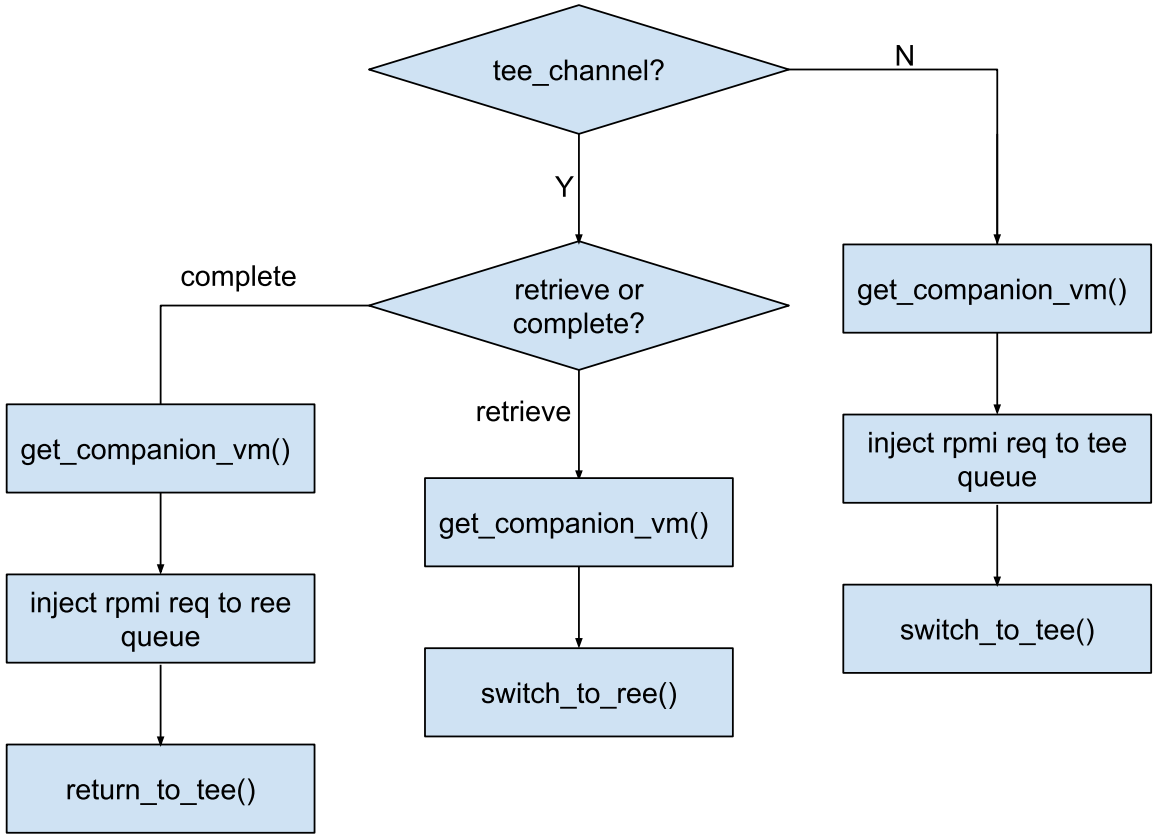
Solution2: TEE on RISC-V (PMP + IOPMP)

RV-ACRN Hypervisor Architecture

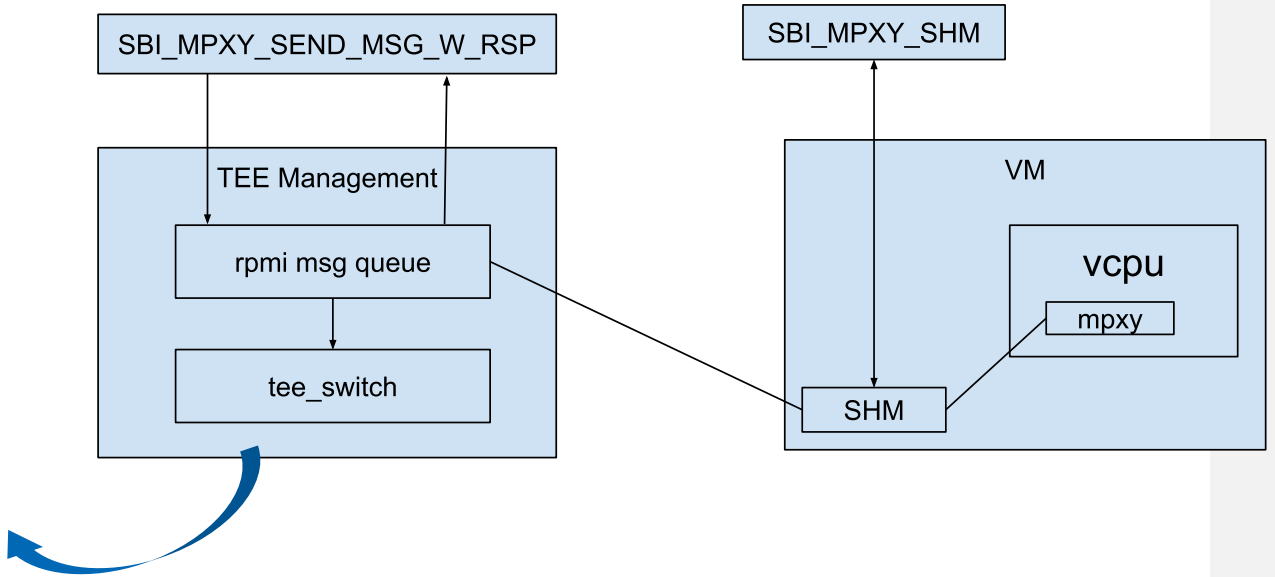


Context Switch Flow

RPMI State Machine

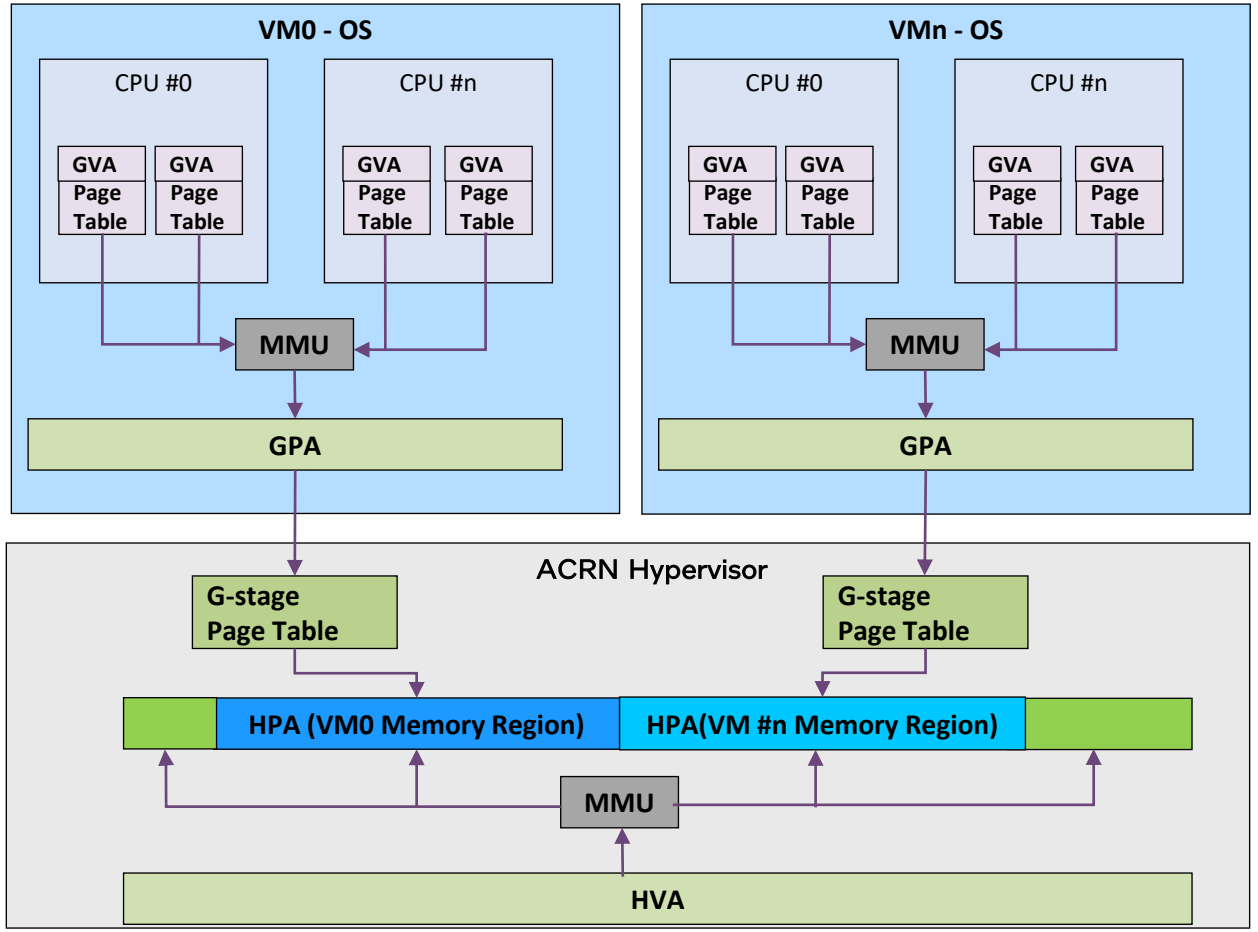


MPXY Structure



Note: TEE always waits for REE request and returns result back to REE after handling the REE's RPMI request. The state machine is driven by RPMI protocol, and the communication channel is setup by SBI_MPXY extension.

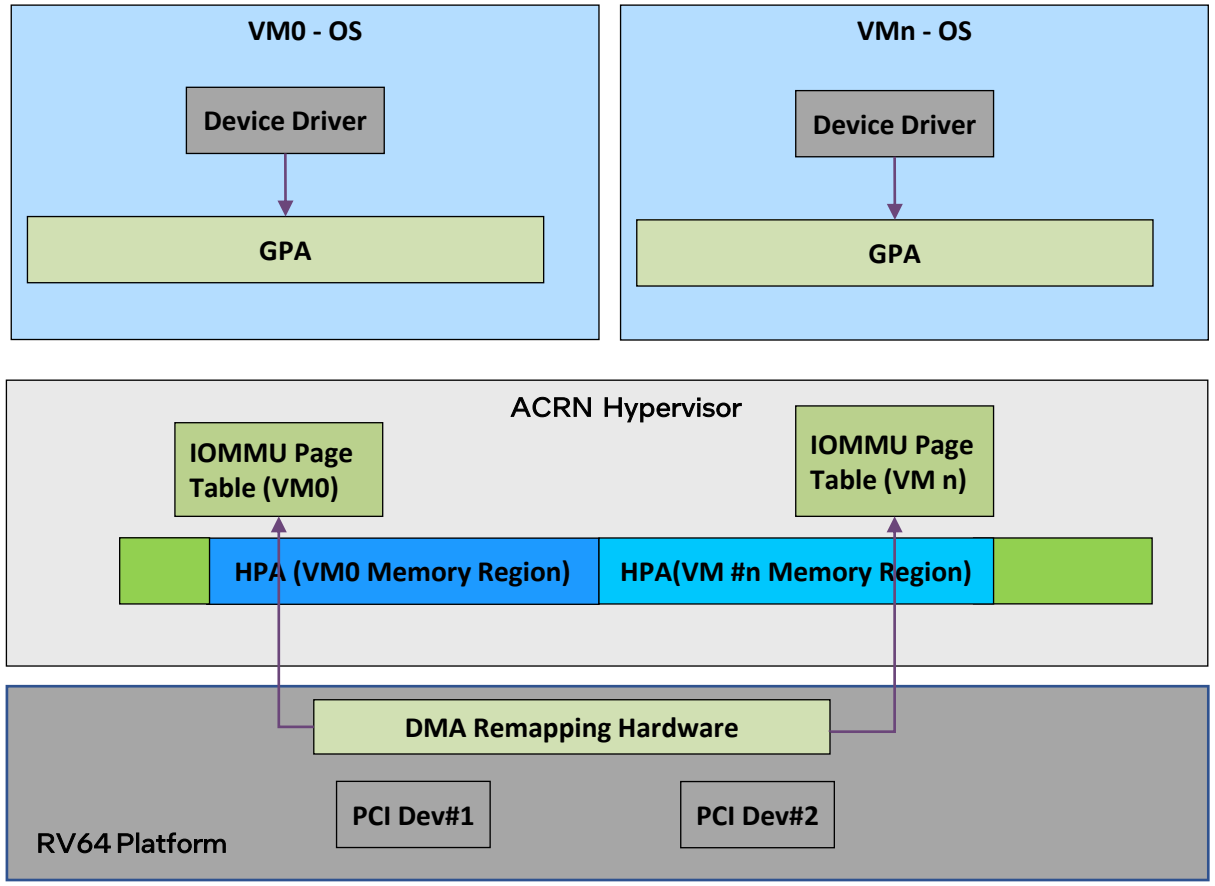
Memory Isolation Between VMs



- RV-ACRN maintains per-VM G-stage page table for each VM to isolate the memory b/w VMs.
- VM can only access predefined memory region defined by G-stage page table.
  - GVA: Guest Virtual Address
  - GPA: Guest Physical Address
  - HPA: Host Physical Address
  - HVA: Host Virtual Address
- Address Translation
  - GVA to GPA via VS-stage page table in guest (vsatp)
  - GPA to HPA via G-stage page table in hypervisor (hgatp)
  - HVA to HPA via MMU page table in hypervisor (satp)

Note: for mACRN mode, PMP is used to isolate the VM memory space, so the G-stage page table is replaced with PMP configurations, which still fits in the same architecture.

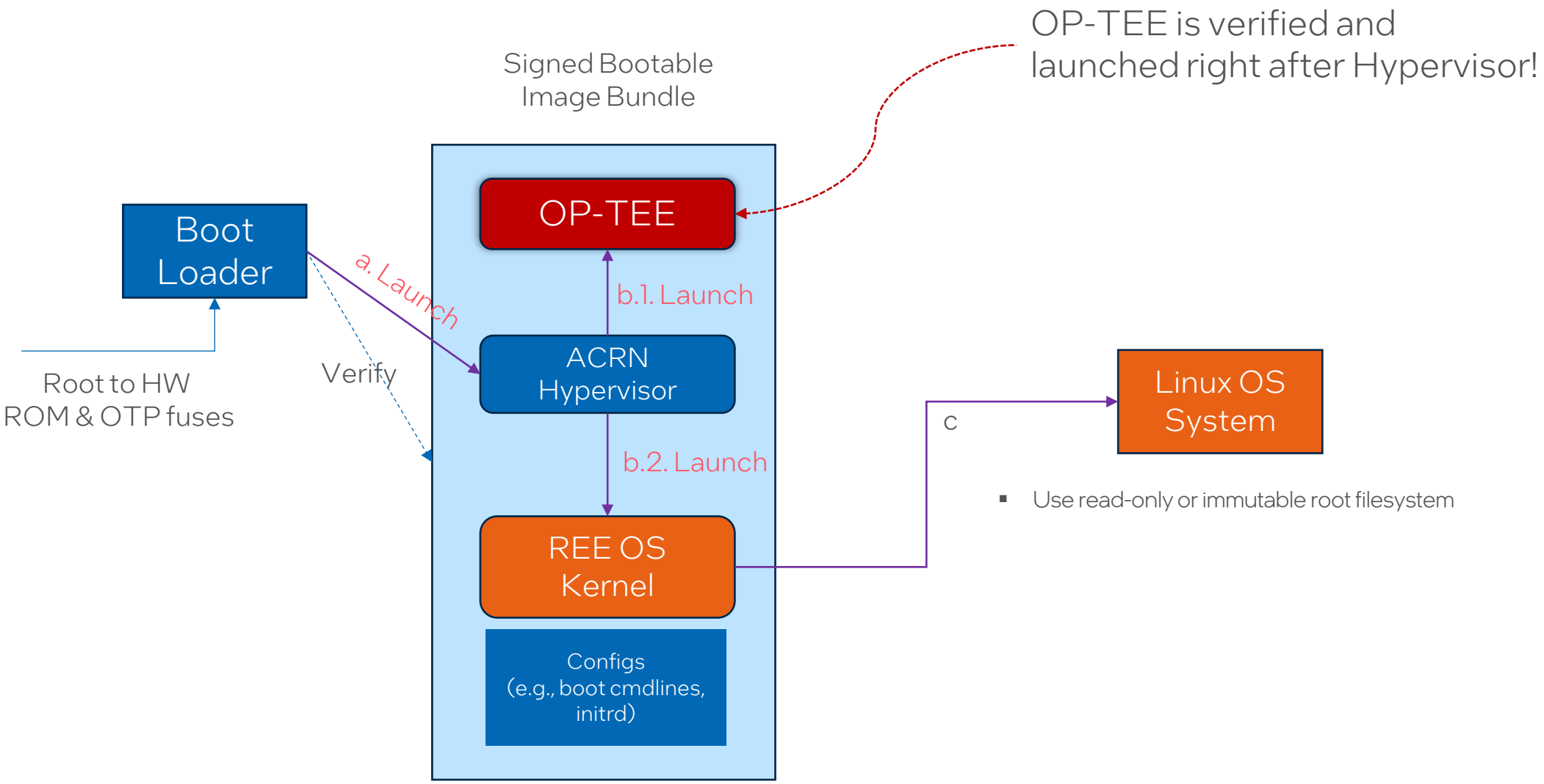
Device DMA Access Isolation with IOMMU



- Driver in guest uses guest physical address(GPA) as DMA target address
- Device issue DMA access using address setup by guest (which is GPA)
- All DMA transactions are captured by IOMMU hardware do the translation from GPA to HPA according to the Address Translation Structures (2nd-stage PT is used on ACRN)
  - With 2nd-stage PT setup for VMs in Hypervisor, Devices assigned to one VM can't access the memory belonging to hypervisor or other VMs via DMA.

Note: for mACRN mode, IOPMP is used to isolate the DMA transactions, so the 2nd-stage PT is replaced with IOPMP configurations, which still fits in the same architecture.

TEE Secure Boot Flow



References & Acknowledgements

- RV-ACRN TEE Repos:
  - RV-ACRN: https://github.com/intel/acrn-riscv/tree/tee-dev
  - REE Linux VM: https://github.com/intel/linux-riscv/tree/hvp-tee
  - OP-TEE VM: https://github.com/haicheng-li/optee_os

- Project ACRN™ Community
  - High Level Security Design (link)

- RISE Security Software WG
  - OPTEE_00_01 - OP-TEE support (link)

- RVI Community
  - Introduction to OP-TEE for RISC-V (link)
  - RISC-V SBI MPXY Spec (link):
  - RISC-V RPMI Spec (link):
  - RISC-V Security Model (link)

Contact Us: Haicheng Li <haicheng.li@intel.com>

Thanks