From RustVMM to Kata-Containers: Securing Container Workloads with RISC-V H-ext Based Virtualization Software

Ruoqing He, Sheng Qu and Yanjun Wu
Institute of Software, Chinese Academy of Sciences

Introduction

As the computing industry moves toward more secure infrastructures, RISC-V presents a unique opportunity with its open-source instruction set architecture (ISA) and its potential to be tailored for security-sensitive applications. The Hypervisor Extension, AIA and IOMMU Spec of RISC-V provides a foundation for hardware accelerated virtualization.

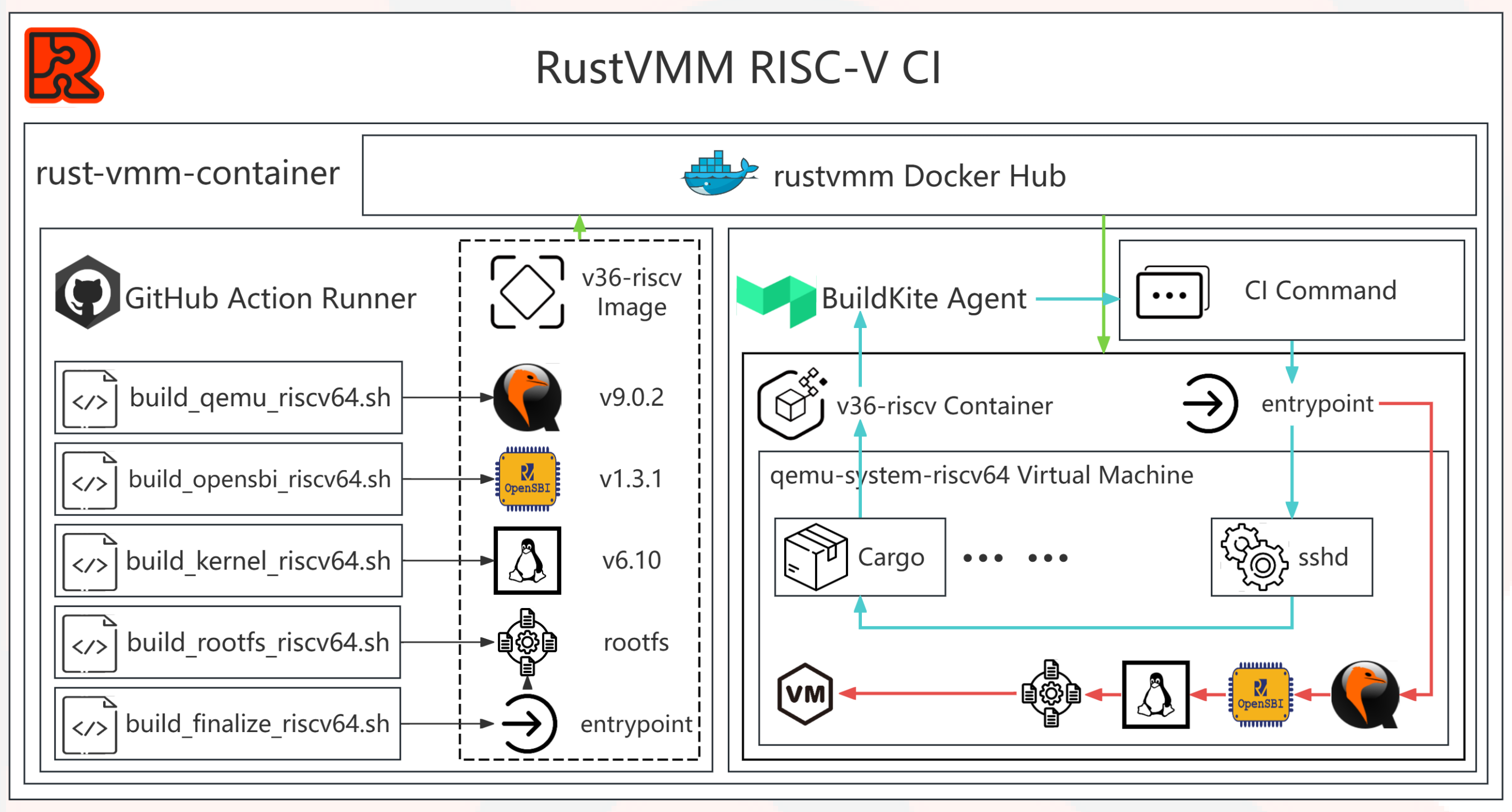
We propose a RustVMM-based virtualization software stack, integrating these to enable the development of lightweight hypervisors (Cloud-Hypervisor, Dragonball, Firecracker, StratoVirt and etc.) and Kata-Containers on RISC-V.

Our objective is to construct a complete Rust virtualization software stack for future RISC-V SoCs compliant with the RVA23 specification and server platform standards, utilizing H Ext. to provide strong isolation, AIA to enable MSI delivering and speed up interrupt handling.

We developed and upstreamed ahead of RISC-V hardware availability, trying to ensure "plug-and-play" readiness of this software stack for future chips.

RustVMM

Due to the absence of RISC-V SoCs with both AIA and IOMM, we are facing great challenges while trying to upstream our work, since there is no real hardware we could use for integrating into communities' CI infrastructure. We managed to address this problem by using QEMU to provide full-emulated RISC-V Virt board with AIA and IOMMU in place to illustrate our works are theoretically correct, and get RISC-V code to merge and evolve with other architectures.

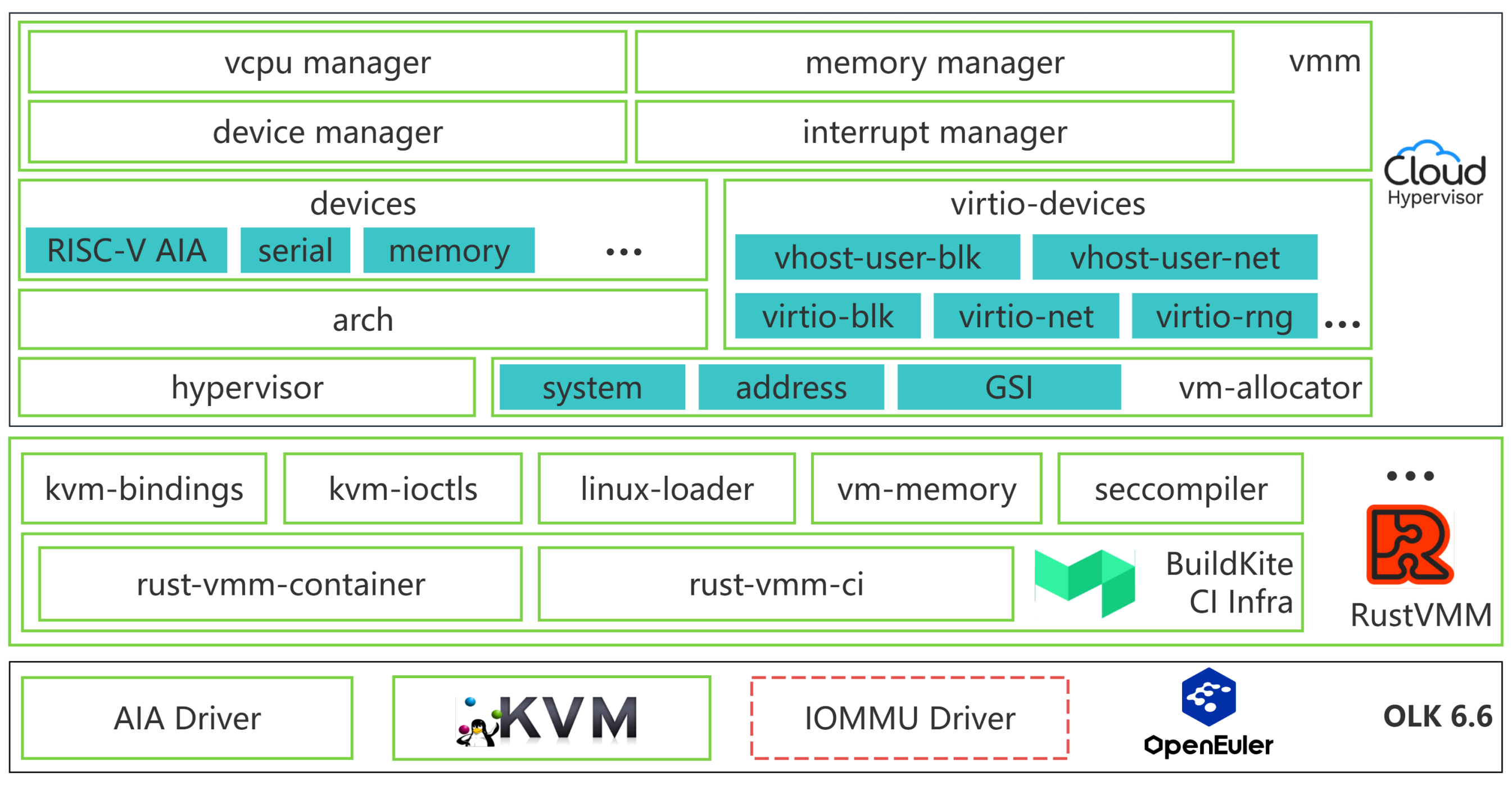


Currently, we have successfully introduced RISC-V architecture support for RustVMM components and facilitated the upstream merging of relevant code to achieve full compatibility with the RISC-V architecture. The RISC-V CI for RustVMM community was officially launched on September 2, 2024.

Since September 23, core repositories such as kvm-bindings and kvm-ioctls have successively released new versions with RISC-V support. This work establishes a critical technical foundation for virtualization software development on RISC-V, and makes RISC-V the 3rd officially supported architecture (the others are x86 & arm) of RustVMM.

Cloud-Hypervisor

Cloud-Hypervisor here as an outstanding hypervisor which is capable of working with Kata-Containers is supported on RISC-V architecture. The overall structure of Cloud-Hypervisor is refactored along the process of supporting RISC-V. After its v45 release, Cloud-Hypervisor is RISC-V enabled and Kata-Containers RISC-V integrated.

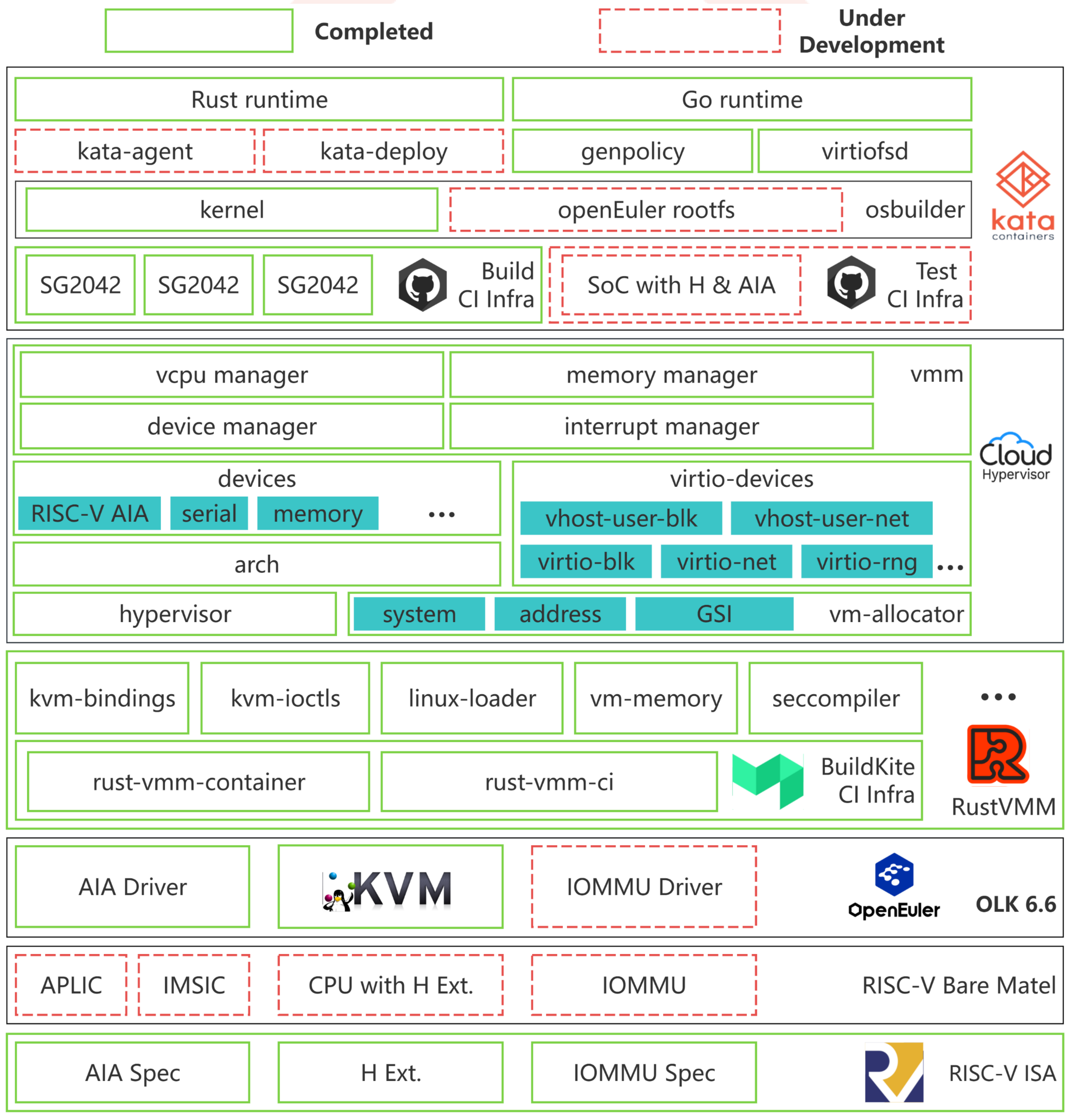


The above secure container software stack owe its security primarily to RISC-V H extension. Even if SoCs with H extension and other features specified in RVA23 profile have not yet come out, we will continue to expand the virtualization software ecosystem of RISC-V, and explore the ways it could be used in secure containers. These works will be firstly tested, verified and distributed on openEuler, and extended to other distributions.

Kata Containers

As shown in below, we follow the principle of "upstream first". We have submitted our work (marked with green boxes in openEuler, RustVMM, Cloud-Hypervisor and Kata Containers) to communities concerned, and most of them are accepted. The rest is still under heavy development and should be ready soon. Kata Containers leverage a hypervisor-based architecture to establish secure container infrastructure.

`



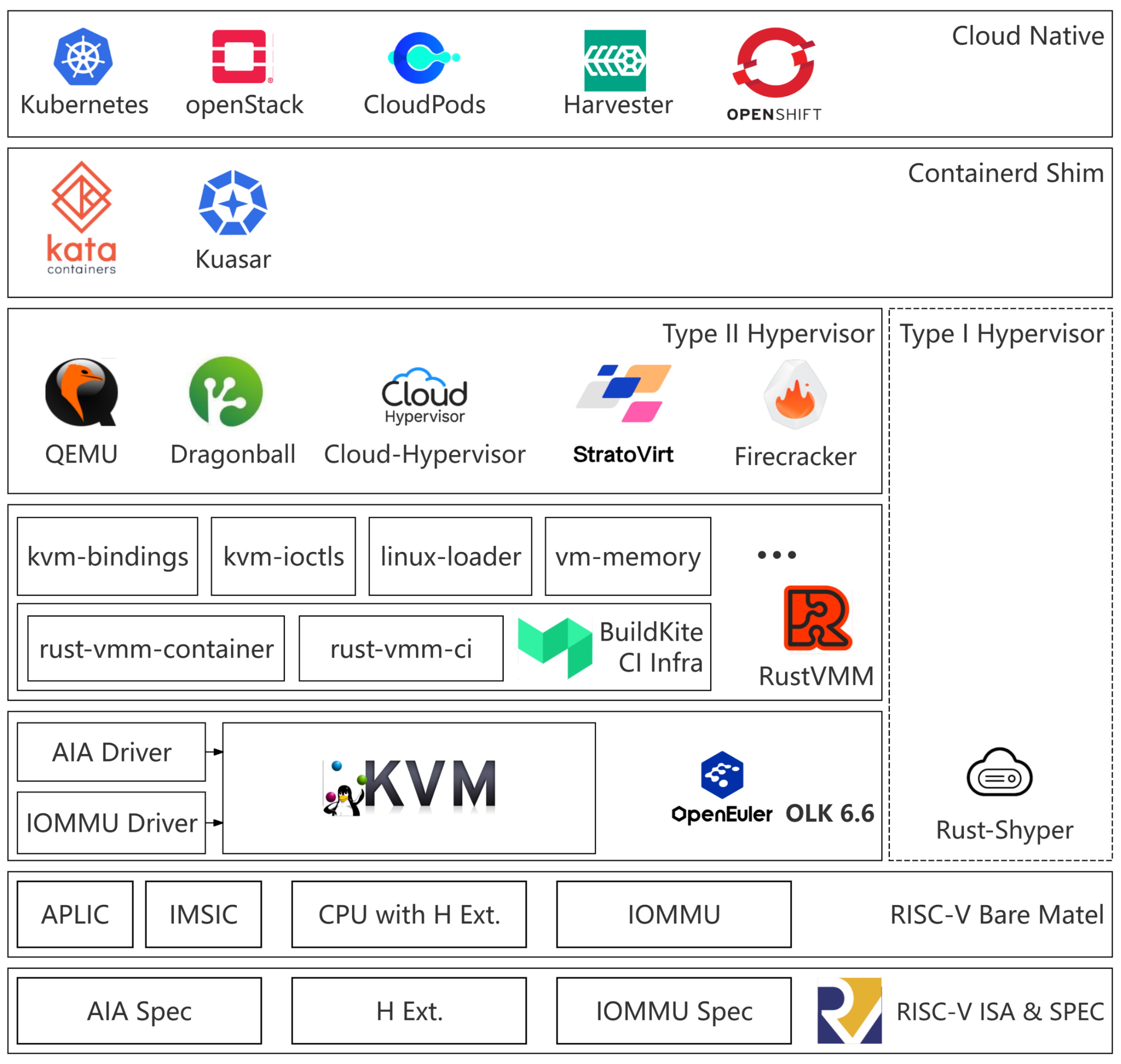
This design significantly strengthens container security boundaries through hardware-enforced isolation while maintaining seamless compatibility with traditional container ecosystems. Such an architecture enables the team to build cloud-native standards-compliant secure virtualization layers for RISC-V based on Kata Containers.

This approach not only enhances RISC-V's competitiveness in critical workload scenarios but also provides an extensible technical foundation for unified management across heterogeneous architectures.

Overview

We present a roadmap of extending the ecosystem of secure container on RISC-V:

- Integrate hypervisors to Kata Containers, support components of Kata Containers to work on RISC-V.
- Expand ecosystem of RISC-V type II hypervisors.
- Extend RustVMM to RISC-V to enable development of Rust Hypervisors on RISC-V.
- Maintain, backport and test new features introduced in mainline Linux kernel to OLK 6.6 of openEuler RISC-V.



This end-to-end stack bridges RISC-V's hardware-assisted virtualization capabilities to cloud-native ecosystems, delivering a secure-by-design alternative to x86/ARM. By aligning with evolving specifications and emerging hardware advancements, we empower RISC-V to secure container, hybrid online-offline orchestration and fault isolation scenarios, and lay down the fundamental of confidential containers for future confidential computing on RISC-V.

Github: https://github.com/RuoqingHe
E-mail: heruoqing@iscas.ac.an