

Linux Plumbers Conference 2023



Contribution ID: 37

Type: **not specified**

KVM

KVM (Kernel-based Virtual Machine) enables the use of hardware features to improve the efficiency, performance, and security of virtual machines created and managed by userspace. KVM was originally developed to host and accelerate “full” virtual machines running a traditional kernel and operating system, but has long since expanded to cover a wide array of use cases, e.g. hosting real time workloads, sandboxing untrusted workloads, depriving third party code, reducing the trusted computed base of security sensitive workloads, etc. As KVM’s use cases have grown, so too have the requirements placed on KVM and the interactions between it and other kernel subsystems.

<p>The KVM Microconference will focus on how to evolve KVM and adjacent subsystems in order to satisfy new and upcoming requirements. Potential topics include:

 Serving inaccessible/unmappable memory for KVM guests (protected VMs); fine-grain permission updates of IOMMU and MMU page tables Optimizing `mmu_notifier`s, e.g. *reducing TLB flushes and spurious zapping* Improving and hardening KVM + perf interactions Implementing arch-agnostic abstractions in KVM (e.g. *MMIO*) Utilizing “fault” injection to increase test coverage of edge cases KVM vs VFIO (e.g. *memory types, a rather hot topic*) Persistence of guest memory and kernel data structure (e.g. *IOMMU page tables*) across `kexec` for live update

<p>Key Attendees: Paolo Bonzini (pbonzini@redhat.com), KVM Maintainer Sean Christopherson (seanjc@google.com), KVM x86 Co-Maintainer Alexander Graf (graf@amazon.de) James Gowans (jgowans@amazon.com) Mickaël Salaün (mic@digikod.net)

Primary authors: BONZINI, Paolo (Red Hat, Inc.); Mr CHRISTOPHERSON, Sean (Google)

Presenter: GOWANS, James (Amazon EC2)

Track Classification: LPC Microconference Proposals