# Unifying KVM API for protected VM and utilities

Isaku Yamahata <isaku.yamahata@intel.com>

November 2023, Linux Plumbers

# Introduction

- KVM confidential guest is expanding
  - SEV-SNP and TDX extending its specific KVM API under KVM_MEMORY_ENCRYPT_OP
  - pKVM is also coming
- The userspace VMM and related tools (debugger, test cases, etc…)has its own enhancement

=>

- Nice to unify KVM API and reduce the difference of userspace VMM or tools
- New ioctl? Sysfs? Debugfs?

# API candidates for consolidation

- Feature enumeration: Technology dependent
  - system wide device ioctl? Sysfs?
- Guest creation: Additional API to the default guest creation
  - VM initialization (vm ioctll): technology dependent parameter
  - vcpu initialization (vcpu ioctl): no parameter
  - Set initial guest memory image and finalize
- Guest destruction
  - The destruction cost of KVM MMU page table can be high. Hint for the user space to tell the guest will destruct and won't run.
  - Parallel destruction by vcpu thread?

# API candidates for consolidation(cont.)

- Guest debug
  - Inspect guest memory
    - API for VM or guest memfd?
  - Inspect guest registers
  - Set break point
- Stats
  - Sysfs or debugfs
- guest<->host communication
  - vsock, technology dependent way, define common way(hypercall/ioio/mmio VMM-agnostic)