Devices in VMs, Again

Luca Abeni

`luca.abeni@santannapisa.it`

April 6, 2020

I/O Devices in VMs

- Hypervisor-based VM (hardware virtualization): what to do for I/O devices?
- Various kinds of solutions $\rightarrow$ various trade-offs between performance, security, transparency of the virtualization, flexibility, ...
  - Exact emulation of existing physical devices
  - Paravirtualization
  - Device pass-through
- Let’s look at the pros and cons of each one...

Emulating Existing I/O Devices

- Complex emulation (resulting in a lot of code)
- Generally bad performance (devices not designed for VMs)
- Support for a lot of different guests
- Standard, well-devined hardware interfaces

Paravirtualized I/O Devices

- Specifically designed for VMs
- Good performance (designed to reduce the number of VM exits)
- Simple host/guest interaction (simpler code)
- The guest must be aware of the VM
- Need for special device drivers in the guest
- Risk to have a lot of non-standard interfaces

I/O Devices Passthrough

- Crazy idea: give the guest access to the host’s physical devices!
  - Is this/can this be safe?
- No need for special device drivers
- No need for device drivers in the hypervisor/host
- Devices are dedicated to a single VM!

Device Passthrough and Security

- Remember? A VM must provide isolation...
- Can device pass-through break this?
  - Think about I/O devices that can act as bus masters...
  - The device accesses physical addresses, not virtual ones!
  - Hence, it can be programmed to read/write outside of the VM address space!!!
- This is not a new issue: μkernel systems already faced similar problems
  - Again, notice the parallelism and similitudes between μkernels and hypervisors...

Safe Device Passthrough

- How to safely give to a guest access to a host's physical device?
- Need for a hardware solution: IOMMU
- Allow to somehow control the accesses of bus mastering devices to physical memory
- The host/hypervisor is in charge of properly configure the IOMMU
- At this point, the guest can safely access an I/O device at native speed!
    - Again, notice that the device is reserved to the guest