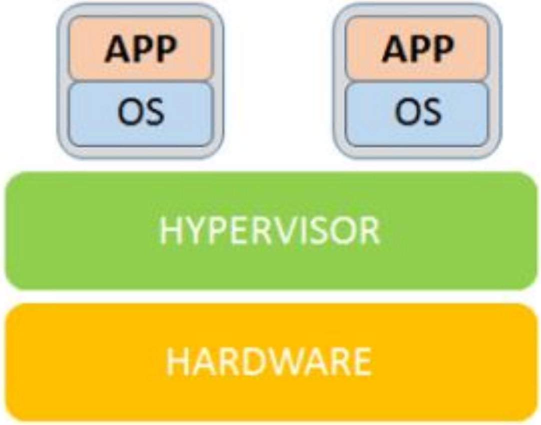
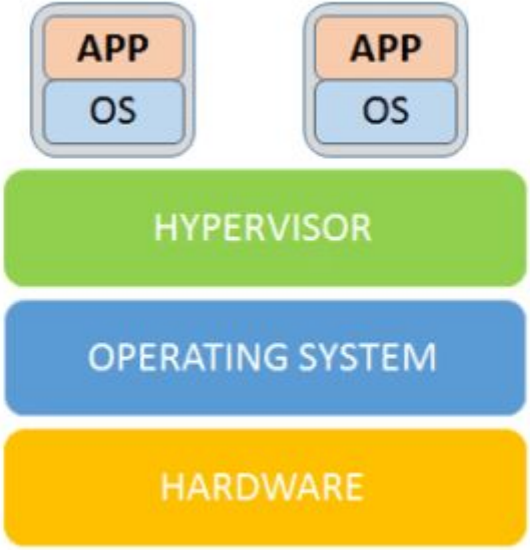
KVM

Darko Bozhinoski,
Ph.D. in Computer Science
Email: D.Bozhinoski@innopolis.ru

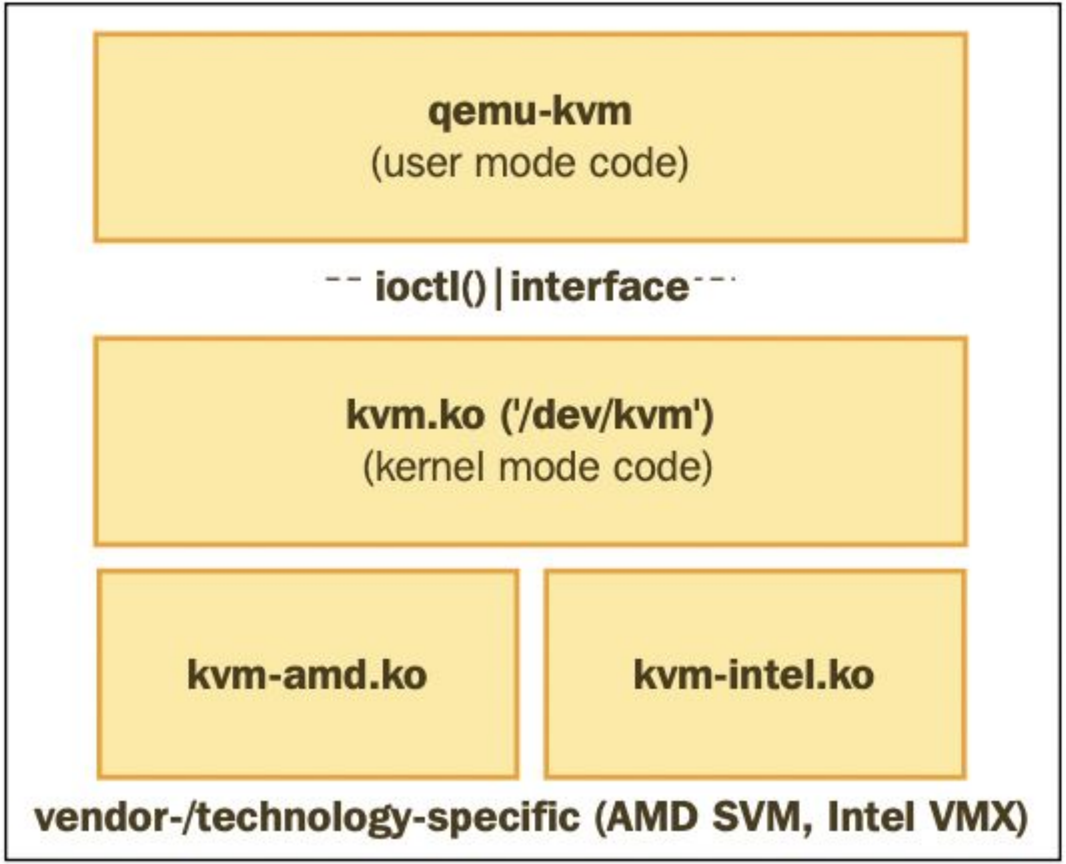
Two types of hypervisors



Type 1 Hypervisor



Type 2 Hypervisor



qemu-kvm

(user mode code)

-- ioctl() | interface ---

kvm.ko ('/dev/kvm')

(kernel mode code)

kvm-amd.ko

kvm-intel.ko

vendor-/technology-specific (AMD SVM, Intel VMX)

LIFE CYCLE OF VMM SOFTWARE

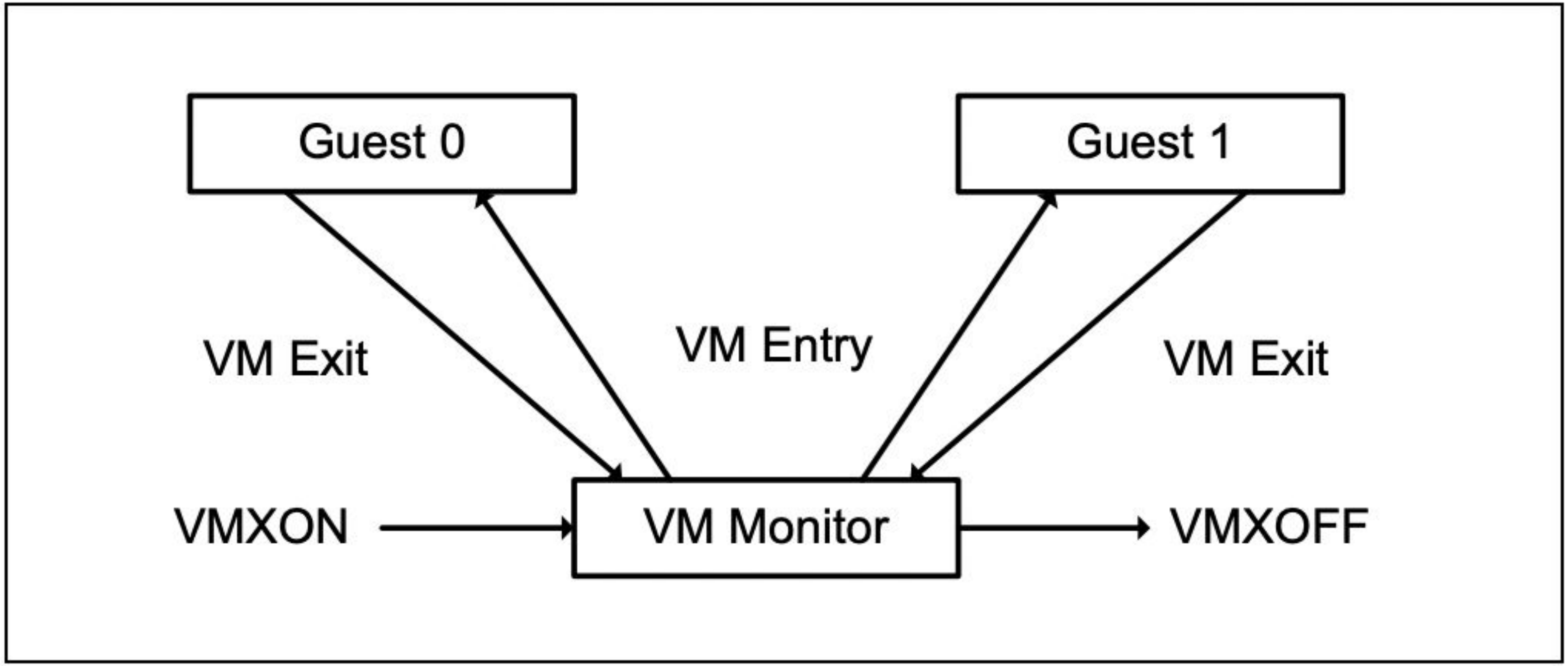


Figure 23-1. Interaction of a Virtual-Machine Monitor and Guests

KVM API

The kvm API is a set of system calls (ioctls) that are issued to control various aspects of a virtual machine. The ioctls belong to the following classes:

- **System ioctls:** These query and set global attributes which affect the whole kvm subsystem. In addition a system ioctl is used to create virtual machines.
- **VM ioctls:** These query and set attributes that affect an entire virtual machine, for example memory layout. In addition a VM ioctl is used to create virtual cpus (vcpus) and devices.
  VM ioctls must be issued from the same process (address space) that was used to create the VM.
- **Cpu ioctls:** These query and set attributes that control the operation of a single virtual cpu.
  vcpu ioctls should be issued from the same thread that was used to create the vcpu, except for asynchronous vcpu ioctl that are marked as such in the documentation.
- **Device ioctls:** These query and set attributes that control the operation of a single device.
  Device ioctls must be issued from the same process (address space) that was used to create the VM.

Assignment 2:

Find out how KVM (kernel virtual machine) does the following:

- Main execution cycle
- #PF(page fault) processing
- entry and exit point of the guest.

Intel Software Developer’s manual, vol 3c, chapter 23:
http://elixir.free-electrons.com/linux/v3.18.53/source

Create a small report where you specify:

- a function where the looping is done, the code line after that guest starts, where it exits and the code line where VMM starts after guest return.

- a common place for handling all page faults in VT-x

You should also describe how you've came to the conclusions you have demonstrated. I expect you to show me the call chains.