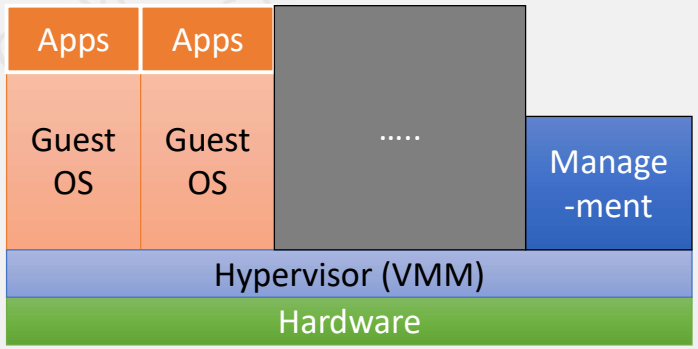
CLOUD COMPUTING APPLICATIONS

Virtualization: Full Virtualization

Prof. Reza Farivar

Full Virtualization

- The virtual machine simulates enough hardware to allow an unmodified "guest" OS (one designed for the same CPU) to be run in isolation
- The virtual machine looks and feels exactly like a real computer, up to the point where a guest operating system cannot tell the difference
- Examples:
  - VirtualBox
  - Virtual PC
  - VMWare
  - QEMU



Virtualization: Privileged and non-privileged instructions

- 1974 paper by Goldberg and Popek described criteria to make a system virtualizable
  - Trap and Emulate
- Safe Instructions
- Unsafe (sensitive) instructions
  - Privileged instructions a subset of unsafe
  - Privileged instruction should cause a trap
- The original X86 was not virtualizable according to the above paper
  - 17 unsafe instructions that were not privileged
    - Intel VT-x and AMD-V later made these privileged

Trap and Emulate

- The classical way to implement a hypervisor is using the "trap and emulate" approach
  - This approach was used by the very first hypervisor developed by IBM in the late 60s
    - IBM System 370
    - Used again today on 64-bit Intel and AMD systems
- The approach usually has good performance, because the majority of the instructions will not cause a trap, and will execute straight on the CPU with no overhead.

Trap and Emulate

- Executable code from the guest can execute directly on the host CPU by the hypervisor
- the hypervisor configures the CPU in such a way that all potentially unsafe instructions will cause a "trap”
- An unsafe instruction is one that for example tries to access or modify the memory of another guest.
- A trap is an exceptional condition that transfers control back to the hypervisor.
- Once the hypervisor has received a trap, it will inspect the offending instruction, emulate it in a safe way, and continue execution after the instruction