CHAPTER 18

Virtual Machines

Exercises

18.1 Describe the three types of traditional hypervisors.

Answer:

a. Type 0—implemented by firmware; low overhead but generally fewer features. Other VMMs can run as guests.

b. Type 1—special-purpose software or general-purpose operating system that provides means to run guests. Takes advantage of available hardware assistance; most feature rich.

c. Type 2—application providing guest execution unbeknownst to the operating system. More overhead and fewer features than type 1.

18.2 Describe four virtualization-like execution environments, and explain how they differ from "true" virtualization.

Answer:
Paravirtualization—guest operating system is modified to perform better and integrate better with the VMM. Does not exactly duplicate native hardware. Programming-environment virtualization—guests are programs written in the programming language specific to the environment. Does not duplicate any real hardware but rather provides an idealized system designed for the language. Emulation—translates each instruction from a different CPU architecture to the current system's instructions. Runs too slowly to meet the standard definition of virtualization. Application containment—does not virtualize underlying hardware, but rather creates an environment for processes with many of the features of virtualization

18.3 Why are VMMs unable to implement trap-and-emulate-based virtualization on some CPUs? Lacking the ability to trap and emulate, what method can a VMM use to implement virtualization?

Answer:

a. If a CPU does not cause a trap if a privileged instruction is run in user mode, the trap-and-emulate method cannot work. For example, an instruction could perform a subset of its function in user mode and a full set in kernel mode. Such an instruction in guest mode does not cause a trap to the VMM in order to have its equivalent kernel mode effect.

b. The more complex and higher overhead binary translation method can be used in such cases.

18.4 What hardware assistance for virtualization can be provided by modern CPUs?

Answer:

VMMs in hardware provide CPU state storage of guests, access to nested page tables can be accelerated via page-table walking hardware, hardware-assisted DMA allows hardware to transfer data directly into a guest memory space, and protection domains can limit which memory can be accessed by each guest for instances such as I/O data transfer.