

Entrega 13

Sistemas Operativos II

Mellino, Natalia

Farizano, Juan Ignacio

Ejercicio 1

- a) **Verdadero:** los dispositivos que típicamente generan más actividad de entrada y salida pueden ser atendidos por drivers paravirtuales.
- b) **Falso:** los contenedores no presentan un hardware virtual, si no que más bien es una virtualización a sistema operativo por lo que no se virtualiza la memoria, si no que funcionan más como si fueran un proceso.
- c) **Verdadero:** los hipervisores proveen más una virtualización a nivel hardware, por lo que para el sistema operativo huésped es como si se estuviera ejecutando directamente sobre una máquina real.
- d) **Verdadero:** este esquema en general resulta más rápido que los hipervisores ya que las instrucciones se cachean y no se salta a modo kernel tan seguido.
- e) **Falso:** para estas máquinas virtuales, por ejemplo Java, no solo alcanza con emular las instrucciones del procesador, si no que también necesitan emularse todos los chips de apoyo, cosa que el hipervisor de tipo I no provee.

Ejercicio 2

Con respecto a las instrucciones sensibles, no conviene hacer que formen parte de los bloques porque generalmente para su ejecución se requieren privilegios especiales y al ejecutarse el S.O. huésped en modo usuario, se debería interrumpir así el S.O. anfitrión ejecuta estas instrucciones en modo kernel.

Por otro lado, con los saltos tampoco es conveniente ya que puede ocurrir que al producirse uno, es difícil predecir en qué bloque se encontrará las siguientes instrucciones a ejecutar después del salto.

Ejercicio 3

- **Hipervisores de tipo I:** esta tecnología podría ayudar ya que al proveer una virtualización a nivel hardware, se proveen las instrucciones y abstracciones necesarias para probar el kernel.
- **Paravirtualización:** esta tecnología podría ayudar dependiendo de lo que se quiera probar. Como es necesario modificar el kernel para que ciertas llamadas a hardware se realicen a través de una API para acceder al hardware simulado en vez de utilizar las llamadas a hardware real (que pueden ser simuladas), es posible que este no sea el entorno de uso en el que se quiera probar el kernel. Sin embargo, en caso de que se quiera probar esta modificación en este entorno, si puede llegar a ser de utilidad.
- **Contenedores:** al ser esta tecnología una virtualización a nivel sistema operativo, no se proveen las instrucciones y llamadas necesarias a nivel hardware que necesita un kernel para funcionar correctamente.