

Virtualización de Servidores:

Cursos Linux Para Ingeniería

Luis Garreta

luis.garreta@javerianacali.edu.co

Ingeniería de Sistemas y Computación
Pontificia Universidad Javeriana – Cali

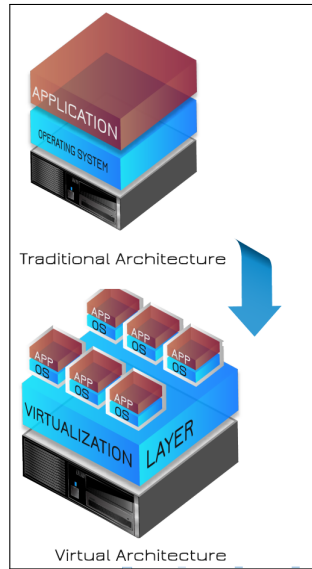
11 de mayo de 2018

Contenido

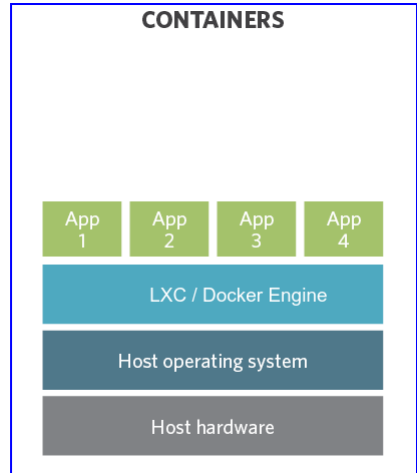
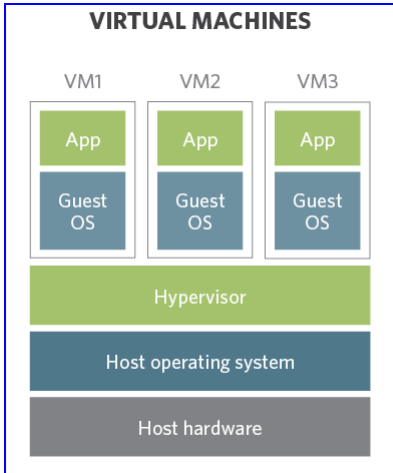
- ▶ Qué son y cómo funcionan los sistemas de virtualización y las herramientas relacionadas.
- ▶ Algunas de las soluciones de virtualización disponibles en las distribuciones de GNU/Linux
- ▶ Explicación de sus características, casos de uso y modo de funcionamiento.

¿Qué es la virtualización?

- ▶ La capacidad de ejecutar en un único equipo físico (el anfitrión o host) múltiples sistemas operativos invitados (guests)
- ▶ Ejecutar programas de usuario dentro de un entorno virtual:
 - ✓ Sin tener que modificarlos, ó
 - ✓ Modificando el sistema operativo invitado para que los programas de usuario puedan funcionar sin cambios
- ▶ Al final, se busca obtener beneficios sobre la configuración física original.



Tendencias de Virtualización



Algunos Términos sobre Virtualización

- ▶ Anfitrión (host)
 - ✓ Es el Sistema Operativo que ejecuta el software de virtualización.
 - ✓ El SO anfitrión controla el hardware real.
- ▶ Invitado ó huésped (guest)
 - ✓ SO virtualizado. Puede haber varios SO invitados en un mismo anfitrión.
 - ✓ Los invitados no deben interferir ni entre ellos ni con el anfitrión.
- ▶ Hipervisor
 - ✓ Software que hace posible la virtualización, se encuentra entre el hardware y el sistema operativo
- ▶ Máquina Virtual:
 - ✓ Software que constituye una Instancia de hardware virtualizado, capaz de cargar en su interior otro sistema operativo
- ▶ Contenedor:
 - ✓ Máquinas virtuales más ligeras que comparten el kernel del SO anfitrión.

Tecnologías de Virtualización

► Hipervisores

- ✓ **KVM:** Kernel Based Virtual Machine:
http://www.linux-kvm.org/page/Main_Page
- ✓ **Xen:** <http://www.xen.org/>
- ✓ **VirtualBox:** <https://www.virtualbox.org/>
- ✓ **VMware Workstation:** http://www.vmware.com/es/products/desktop_virtualization/workstation/overview.html
- ✓ **VMware vSphere Hypervisor (ESXi):**
<http://www.vmware.com/products/server/overview.html>
- ✓ **Microsoft Hyper-V:**
<http://www.microsoft.com/en-us/server-cloud/hyper-v-server/>

► Contenedores:

- ✓ lxc, Docker, OpenVZ y Linux-VServer,

Hipervisor

- ▶ Permite que diferentes SOs, tareas y configuraciones de software coexistan en una misma máquina física.
- ▶ Abstrae y Maneja los recursos físicos de la máquina anfitriona para las distintas máquinas virtuales.
- ▶ Garantizan un nivel de aislamiento entre los invitados.
- ▶ Proporcionan una interfaz única para el hardware.

Tipos de Hipervisores: Dos enfoques

Type 1 Hypervisor



Type 2 Hypervisor



Tipos de Hipervisores

Tipo 1 (“nativo” ó “bare-metal”):

- ▶ Se ejecuta directamente sobre el hardware y gestiona los SOs invitados.
- ▶ Al SO se le llama Dominio de Control y se ejecuta sobre el hipervisor.
- ▶ Los invitados son Dominios Lógicos.
- ▶ Ejemplos:
 - ✓ Xen, Citrix XenServer, KVM, VMware ESX/ESXi, Microsoft Hyper-V.

Type 1 Hypervisor



Tipos de Hipervisores

Tipo 2 (“hosted”):

- ▶ Se ejecuta en el entorno convencional de un SO.
- ▶ Representa una capa software que se ejecuta sobre el SO anfitrión.
- ▶ Ejemplos:
 - ✓ VMware Workstation, VMware Server, VirtualBox, QEMU, Microsoft Virtual PC.

Type 2 Hypervisor



Clases de Virtualización

- ▶ Virtualización de recursos:
- ▶ Virtualización de plataforma.

Virtualización por Recursos

- ▶ No muy nombrada como virtualización
- ▶ Es la que involucra la simulación de recursos, como volúmenes de almacenamiento, espacios de nombres y recursos de red:
- ▶ Ejemplos:
 - ✓ Discos RAID y gestores de volúmenes (como Linux LVM)
 - ✓ Virtualización de almacenamiento como SAN (Storage Area Network).
 - ✓ Redes Privadas Virtuales (VPN).
 - ✓ Sistemas multiprocesador y multinúcleo.
 - ✓ Clusters, grid computing, cloud computing

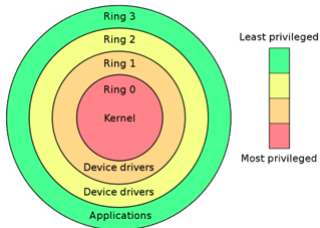
Virtualización de Plataforma

Consiste en la creación de una máquina virtual utilizando una combinación de hardware y software.

- ▶ Se lleva a cabo a través de un software de virtualización o hipervisor (ej. VirtualBox, VMWare, KVM, ...).
- ▶ Este software actúa de host o anfitrión y simula un determinado entorno computacional (máquina virtual).
- ▶ En esta máquina virtual se instala un software guest o invitado, normalmente un sistema operativo completo.
 - ✓ Instalado de la misma manera que si lo estuviera en una máquina real.
- ▶ La simulación debe ser lo suficientemente robusta:
 - ✓ Debe soportar todas las interfaces externas del software invitado, incluidos, en algunos casos, drivers de hardware.

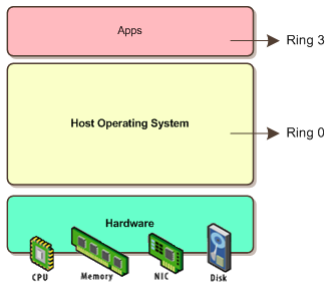
Anillos de Protección

- ▶ Kernel se ejecuta en el anillo 0 (más privilegios)
- ▶ Aplicaciones se ejecutan en el anillo 3 (menos privilegios)



Sistema Tradicional (No virtualización)

- Como es la arquitectura de un sistema tradicional (sin virtualización)

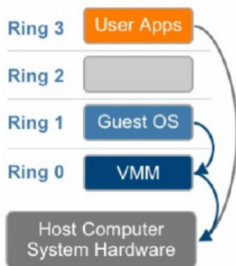


Tipos de Virtualización de Plataforma

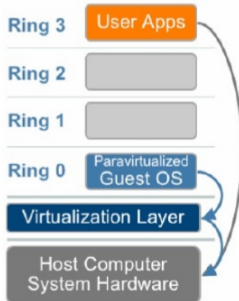
- ▶ Tradicional.
- ▶ Emulación
- ▶ Virtualización nativa o completa.
- ▶ Paravirtualización
- ▶ Virtualización asistida por hardware.
- ▶ Virtualización a nivel de sistema operativo.
- ▶ Otros tipos.

Resumen Tipos de Virtualización

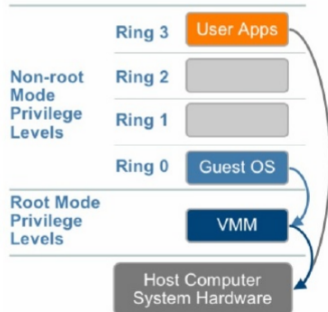
Full Virtualization



Para Virtualization



Hardware Assisted Virtualization



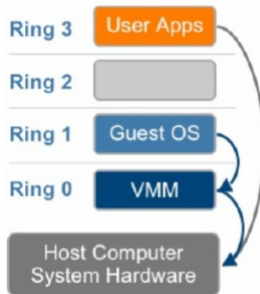
Emulación

- ▶ La máquina virtual simula un hardware completo.
- ▶ La VM admite SOs invitados sin modificar para arquitecturas CPU completamente diferentes a la CPU del SO anfitrión.
- ▶ Un emulador permite ejecutar programas en una plataforma diferente para la que fueron escritos.
- ▶ Ejemplos:
 - ✓ QUEMU: Emulador de máquinas que usa traslación dinámica de instrucciones
 - ✓ Principalmente QUEMU trabaja como un "recompilador" que transforma el código binario de un procesador hacia código de otro procesador distinto (e.g. MIPS, PPC, o ARM en x86PC).

Virtualización nativa o completa

- ▶ La máquina virtual simula un hardware suficiente para poder permitir a un sistema operativo invitado sin modificar, correr de forma aislada sobre el mismo tipo de CPU que la máquina anfitriona.
- ▶ En virtualización nativa, tanto el sistema anfitrión como el sistema operativo invitado se ejecutan sobre la misma CPU.
- ▶ Virtualización completa es importante para ejecutar SO propietarios (ej. Windows) los cuales no se tiene acceso al código para modificarlo.

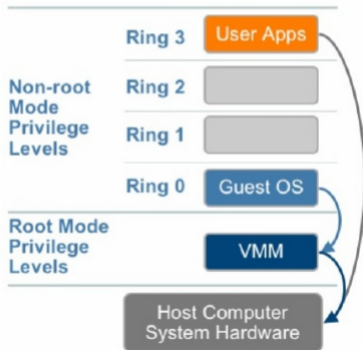
Full Virtualization



Virtualización asistida por hardware

- ▶ Es un caso especial de la virtualización completa en la que se cuenta con ayuda del procesador.
- ▶ Intel con su tecnología VT-x y AMD con AMD-V proporcionan ayuda por hardware al software de virtualización.
- ▶ Estas opciones se tienen que configurar en la BIOS
- ▶ Ejemplos:
 - ✓ KVM, VMware Workstation, VMware Fusion, Microsoft Hyper-V, Microsoft Virtual PC, Xen, Parallels Desktop for Mac, VirtualBox y Parallels Workstation.

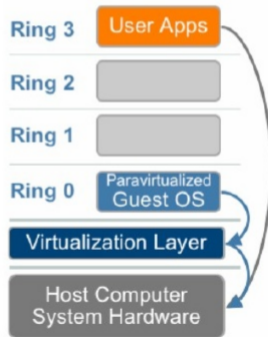
Hardware Assisted Virtualization



Paravirtualización

- ▶ Se modifica el kernel del sistema *invitado* para correr en el hipervisor.
- ▶ El nuevo kernel llama a librerías (API) del hipervisor
- ▶ Las librerías acceden directamente al hardware
- ▶ Las llamadas del sistema operativo invitado al hypervisor se denominan **hypercalls**.
- ▶ Ejemplos: Xen en CPU estándar.

Para Virtualization



Virtualización a nivel de Sistema Operativo

► Virtualización basada en contenedores

► Un contenedor:

- ✓ Es una maquina virtual ligera,
- ✓ Comparte funciones con el kernel del sistema operativo,
- ✓ SO esta realizando las funciones de hipervisor

► Ejemplos:

- ✓ OpenVZ, Linux-VServer, **LXC** (Linux Containers), AIX Workload Partitions, Parallels Virtuozzo Containers, y iCore Virtual Accounts.

