

Taller de Administración de Servidores Linux

CC5308

Clase 10: 31 de Marzo de 2011

Virtualizacion

Eduardo Mercader Orta
emercade [at] nic . cl

Copyright © 2011
Creative Commons 3.0-cl by-nc-sa

Introducción

- Objetivos:
 - Mejor aprovechamiento de los recursos
 - Disminuir el impacto medio ambiental
 - Ahorros en espacio y energía
 - Simplicidad en la operación
 - Distribuir la administración
 - Independizarse del hardware
 - Simplicidad en la migración
 - Mejor tolerancia a fallas

Introducción

- Diferentes enfoques:
 - Virtualización por emulación (QEMU, Bochs)
 - Virtualización completa (Vmware, VirtualBox)
 - Paravirtualización (Xen/UML)
 - Virtualización por hardware (KVM)
 - Virtualización a nivel del sistema operativo (FreeBSD jail)
 - Virtualización a nivel de bibliotecas (Wine)
- Algunas controladas por un supervisor (Hypervisor)

Emulación

- QEMU, emula un x86 sobre hardware diferente: x86, PowerPC, SPARC, MIPS, ARM
- Cada instrucción del hardware emulado es ejecutada por el hardware real
- Se emulan todos los dispositivos necesarios: memoria, periféricos, red.

Virtualización completa

- Vmware y Virtualbox, proporcionan un supervisor (Hypervisor), que intermedia entre el Sistema operativo instalado y el hardware real.
- Requiere que el sistema operativo a ejecutar esté diseñado para el hardware real
- Debe traducir las instrucciones del sistema operativo virtualizado.
- Su rendimiento es mayor cuando el hardware soporta virtualización.
- Disponibles para múltiples sistemas operativos.

Paravirtualización

- El sistema operativo virtualizado accede directamente al hardware, requiere kernel modificado para operar, pues el supervisor colabora con el S.O.
- Xen, opera sobre Intel VT o AMD Pacífica. Es una solución de código abierto, de amplio uso empresarial.
- UML (User Mode Linux), modo en el que el kernel de linux permita ejecutar a otros kernel linux. Disponible desde kernel 2.6. Se comparten dispositivos.

KVM

- Corresponde a una emulación completa
- Un módulo en el kernel (kvm), permite virtualizar los dispositivos del anfitrión.
- El invitado utiliza QEMU (en modo no emulado), para la operación del hardware.
- Permite ejecutar linux 32 y 64 bits y windows

FreeBSD Jail

- Corresponde a varios sistemas operando en modo compartido en un anfitrión.
- Cada invitado tiene sus propios procesos, archivos, usuarios y superusuarios
- Los invitados no puede acceder a sus vecinos
- Los procesos en los invitados son adjuntados al kernel de cada anfitrión para limitar sus privilegios.
- Todos los invitados comparten el mismo kernel

Virtualización por bibliotecas

- Wine implementa la API de Win32 en Linux.
- Se permite ejecutar aplicaciones Windows en Linux, simulando al sistema operativo original.
- Las aplicaciones piensan que están en su ambiente nativo
- No es exactamente una “virtualización”, pues no emula. Es una implementación alternativa.

Seguridad

- Los diferentes esquemas de virtualización proveen diferentes mecanismos de seguridad.
- En general permiten aislar los diferentes invitados entre ellos
- Las debilidades del anfitrión serán una debilidad para todos los invitados.

Límites

- Cada virtualización provee diferentes límites
 - Cantidad de procesadores emulados
 - Cantidad de memoria
 - Cantidad de invitados soportados

Actividad Personal

- Identificar el tipo de virtualización que soporta su equipo: `egrep '(vmx|svm)' /proc/cpuinfo`
- Instalar alternativamente kvm o xen como hypervisor
- Crear una máquina virtual (virt-manager)
- Interactuar con ella por medio de virsh y/o xm:
 - Conectarse a la consola
 - Detenerla
 - Iniciarla
 - Modificar su memoria y cantidad de procesadores
 - Agregar disco