



**UNIVERSIDAD LATINA
DE COSTA RICA**

POWERED BY **Arizona State University**

Sistemas Operativos II

Profesor

Carlos Mendez Rodriguez

Estudiante:

Javier Díaz Mora / 20200120139

14.1

Problemática Tradicional:

- Las aplicaciones solían ejecutarse directamente en un sistema operativo (OS) en una PC o servidor.
- Los proveedores de aplicaciones tenían que reescribir partes de sus aplicaciones para cada OS/plataforma, aumentando el tiempo de lanzamiento al mercado, los defectos, los esfuerzos de prueba y los costos.
- Para soportar múltiples OS, los proveedores necesitaban gestionar y soportar múltiples infraestructuras de hardware y OS, un proceso costoso y que consumía muchos recursos.

Introducción de la Virtualización:

- La virtualización permite que una sola PC o servidor ejecute múltiples OS o múltiples sesiones de un solo OS simultáneamente.
- Un software de virtualización puede alojar numerosas aplicaciones en una única plataforma, soportando múltiples máquinas virtuales (VMs) con características de diferentes OS y hardware.
- La virtualización no es nueva; IBM la ofreció en los años 70. Se popularizó en la década de 2000 con servidores x86 debido al bajo uso de recursos en servidores bajo la estrategia "una aplicación, un servidor".

Ventajas de la Virtualización:

- **Hypervisor:** Software que actúa como intermediario entre el hardware y las VMs, permitiendo que múltiples VMs compartan recursos de un solo servidor físico.
- **Consolidación:** Permite una alta relación de consolidación, reduciendo significativamente el número de servidores físicos, costos de energía, enfriamiento y mantenimiento.
- **Despliegue rápido:** Nuevas VMs se pueden desplegar en minutos.
- **Versatilidad:** Optimiza el uso del hardware al manejar múltiples aplicaciones en una sola máquina.
- **Consolidación y agregación:** Recursos como servidores y almacenamiento pueden ser usados más eficientemente.
- **Dinámica y gestión fácil:** Asignación dinámica de recursos, balanceo de carga y tolerancia a fallos.
- **Mayor disponibilidad:** VMs pueden reiniciarse rápidamente en caso de fallos físicos del servidor.

Uso Comercial y Beneficios:

- Proveedores como VMware y Microsoft ofrecen soluciones de VM ampliamente usadas.
- La virtualización de servidores permite particionar un solo host en múltiples servidores independientes, conservando recursos de hardware y permitiendo la migración rápida de servidores.
- Ha sido clave en aplicaciones de "big data" y en la implementación de infraestructuras de computación en la nube.
- También se usa en entornos de escritorio para ejecutar múltiples OS, como Windows y Linux.

14.2

Introducción a la Virtualización y los Hipervisores:

- La virtualización es una forma de abstracción que permite ejecutar múltiples máquinas virtuales (VMs) en un solo servidor físico. Un hipervisor actúa como un intermediario que gestiona y asigna recursos físicos a las VMs.

Funciones Principales de un Hipervisor:

- **Gestión de Ejecución de VMs:** Programa la ejecución de las VMs, gestiona la memoria virtual para garantizar el aislamiento y realiza el cambio de contexto entre diferentes estados de procesador.
- **Emulación de Dispositivos y Control de Acceso:** Emula todos los dispositivos de red y almacenamiento esperados por las VMs, y controla el acceso a los dispositivos físicos.
- **Operaciones Privilegiadas:** Ejecuta operaciones privilegiadas en nombre de las VMs para garantizar el aislamiento y la seguridad.
- **Gestión del Ciclo de Vida de las VMs:** Configura y controla el estado de las VMs (inicio, pausa, detención).
- **Administración de la Plataforma y Software del Hipervisor:** Configura parámetros de interacción con los usuarios y el software del hipervisor.

Tipos de Hipervisores:

- **Tipo 1:** Se instala directamente sobre el hardware físico y gestiona los recursos físicos directamente. Ejemplos incluyen VMware ESXi, Microsoft Hyper-V, y Xen.
- **Tipo 2:** Se ejecuta como un módulo de software sobre un sistema operativo anfitrión. Ejemplos son VMware Workstation y Oracle VM VirtualBox.

Paravirtualización:

- Técnica de virtualización asistida por software que utiliza APIs especializadas para optimizar el rendimiento de las VMs. Requiere soporte paravirtualizado en el kernel del sistema operativo invitado y controladores específicos.

Virtualización Asistida por Hardware:

- Extensiones de virtualización añadidas por fabricantes de procesadores como AMD-V e Intel VT-x, que mejoran el rendimiento de los hipervisores al realizar ciertas operaciones directamente en el hardware.

Aplicaciones Virtuales (Virtual Appliance):

- Software autónomo distribuido como una imagen de máquina virtual, que incluye un conjunto empaquetado de aplicaciones y un sistema operativo invitado. Puede ejecutarse en hipervisores de tipo 1 o tipo 2.

Aplicaciones de Seguridad (Security Virtual Appliance - SVA):

- Herramienta de seguridad que monitoriza y protege otras VMs desde una VM especialmente endurecida fuera de las VMs de usuario. Utiliza la API de introspección de máquinas virtuales del hipervisor para obtener visibilidad y asegurar la red virtual.

14.3

Hipervisores

Un hipervisor es fundamental para la gestión y orquestación de máquinas virtuales (VMs). Proporciona la abstracción necesaria, aislando y emulando los recursos hardware para las VMs. Los hipervisores se categorizan en dos tipos: Tipo 1 y Tipo 2.

Hipervisores Tipo 1:

- Se instalan directamente en el hardware físico del servidor, similar a un sistema operativo.

- Ofrecen control directo sobre el hardware, lo que conduce a un mejor rendimiento y utilización de recursos.
- Ejemplos: VMware ESXi, Microsoft Hyper-V, variantes de Xen.

Hipervisores Tipo 2:

- Se ejecutan sobre un sistema operativo existente.
- Dependientes del sistema operativo anfitrión para la gestión de recursos, lo que puede implicar mayor sobrecarga y rendimiento inferior en comparación con los hipervisores Tipo 1.
- Ejemplos: VMware Workstation, Oracle VM VirtualBox.

Funciones Clave de los Hipervisores:

- **Gestión de Ejecución:** Maneja la programación, aislamiento y cambio de contexto de las VMs.
- **Emulación y Control de Dispositivos:** Emula dispositivos de red y almacenamiento para las VMs, gestionando el acceso a dispositivos físicos.
- **Ejecución de Operaciones Privilegiadas:** Maneja operaciones privilegiadas en nombre de las VMs invitadas.
- **Gestión del Ciclo de Vida de las VMs:** Controla los estados de las VMs como inicio, parada, pausa y configuración.
- **Administración de Plataforma y Software:** Gestiona las interacciones de usuario y las configuraciones de software del hipervisor.

Virtualización de Contenedores

Contenedores:

- Los contenedores proporcionan entornos aislados para aplicaciones en un kernel de sistema operativo compartido.
- Son más livianos en comparación con las VMs, ya que no requieren un sistema operativo separado para cada aplicación.
- Los contenedores comparten el kernel del sistema operativo anfitrión, reduciendo la sobrecarga y mejorando la eficiencia.

Grupos de Control del Kernel de Linux (cgroups):

- Los cgroups permiten la gestión de recursos y el aislamiento para los contenedores.
- Introducidos en la versión del kernel Linux 2.6.24.
- Sus características incluyen límites de recursos, priorización, contabilidad y control de grupos de procesos.

Motor de Contenedores:

- El motor de contenedores es responsable de configurar y gestionar los contenedores.
- Incluye mantener un entorno de ejecución, gestionar sistemas de archivos y solicitar recursos al kernel.

Ciclo de Vida del Contenedor:

- **Configuración:** Configurar parámetros como redes, sistemas de archivos y límites de recursos.
- **Gestión:** Gestionar operaciones de contenedores como inicio, parada, congelación y migración.

Beneficios Clave de los Contenedores:

- Livianos y eficientes debido al kernel de sistema operativo compartido.
- Despliegue rápido y escalabilidad de aplicaciones.
- Portabilidad mejorada de aplicaciones.

Desafíos con los Contenedores:

- Limitada portabilidad entre diferentes kernels de sistemas operativos.
- Posibles vulnerabilidades de seguridad debido al kernel de sistema operativo compartido.
- Menos adecuados para aplicaciones que requieren configuraciones de kernel únicas.

Docker

Componentes de Docker:

- **Imagen de Docker:** Plantillas de solo lectura para crear contenedores.
- **Cliente de Docker:** Solicita la creación de contenedores.
- **Anfitrión de Docker:** Ejecuta aplicaciones en contenedores.
- **Motor de Docker:** Paquete de tiempo de ejecución para gestionar contenedores.
- **Máquina de Docker:** Instala el motor de Docker y configura clientes.
- **Registro de Docker:** Almacena imágenes de Docker.
- **Docker Hub:** Repositorio público para compartir y almacenar imágenes de Docker.

Docker ha ganado popularidad debido a su facilidad de uso, despliegue rápido y la capacidad de almacenar y recuperar imágenes de contenedores desde un repositorio central, simplificando significativamente el proceso de contenerización.

Comparación: Máquinas Virtuales vs. Contenedores

Máquinas Virtuales:

- **Aislamiento:** Cada VM ejecuta un sistema operativo completo, proporcionando un aislamiento y seguridad robustos.
- **Sobrecarga de Rendimiento:** Mayor sobrecarga debido a la ejecución de instancias separadas de sistemas operativos.
- **Flexibilidad:** Pueden ejecutar diferentes tipos de sistemas operativos en el mismo host.

Contenedores:

- **Livianos:** Comparten el kernel del sistema operativo anfitrión, reduciendo el uso de recursos y mejorando la eficiencia.
- **Despliegue Rápido:** Más rápidos de iniciar y gestionar en comparación con las VMs.
- **Portabilidad:** Capaces de mover aplicaciones rápidamente entre sistemas con kernels de sistema operativo compatibles.

Cada enfoque tiene sus ventajas y limitaciones, haciéndolos adecuados para diferentes casos de uso. Las VMs proporcionan un aislamiento robusto y compatibilidad con diversos entornos de sistemas operativos, mientras que los contenedores ofrecen eficiencia y rapidez, especialmente para aplicaciones basadas en Linux.