

2-6-2018

ILUSTRACIÓN 1: SISTEMA OPERATIVO DE REDES



**Facilitador:** Ing. Manuel Flores Villatoro

**Tema a tratar:** Sistema de virtualización KVM

**Integrantes**

Nº	Apellidos	Nombres	Carnet	Participación
1	Ramos Reyes	Anabel	RR02110122	94%
2	Ramírez Rivera	Apolinario	RR01133447	98%
3	Mejía Flores	José Willíam	MF01132606	100% (Líder)

San Salvador 02 de junio 2018



## Índice

Índice de tablas	3
Índice de Ilustraciones	4
Objetivos de la virtualización	5
Resumen	6
Palabras claves	7
Introducción	8
Marco Teórico	10
MÁQUINAS VIRTUALES DE SISTEMA	11
MÁQUINAS VIRTUALES DE PROCESO	12
Materiales y Métodos (Metodología)	13
KVM	13
¿Cómo funciona KVM?	14
Tipos de virtualización	15
Funciones de KVM	18
Instalación de KVM	21
CUADRO COMPARATIVO ENTRE HERRAMIENTAS DE VIRTUALIZACIÓN	23
Características relevantes: Virtualizadores más importantes	24
Resultados	27
Conclusión	32
Recomendaciones	33
Glosario	34
Hipervisor:	36
Referencias bibliográficas	41

## Índice de tablas

TABLA 1: CUADRO COMPARATIVO	23
TABLA 2: PRESUPUESTO DEL PROYECTO	26

## Índice de Ilustraciones

ILUSTRACIÓN 1: SISTEMA OPERATIVO DE REDES	1
ILUSTRACIÓN 2: TIPOS DE HYPERVISOR	6
ILUSTRACIÓN 3: PALABRAS CLAVES	7
ILUSTRACIÓN 4: SERVIDORES 1	8
ILUSTRACIÓN 5: SERVIDORES 2	8
ILUSTRACIÓN 6: MARCO TEÓRICO - MÁQUINAS VIRTUALES	10
ILUSTRACIÓN 7: MÁQUINA VIRTUAL DE SISTEMA	11
ILUSTRACIÓN 8: MÁQUINA VIRTUAL DE PROCESO	12
ILUSTRACIÓN 9: HYPERVISOR KVM	13
ILUSTRACIÓN 10: KVM	14
ILUSTRACIÓN 11: VIRTUALIZACIÓN DE DATOS	15
ILUSTRACIÓN 12: VIRTUALIZACIÓN DE ESCRITORIOS	16
ILUSTRACIÓN 13: VIRTUALIZACIÓN DE SERVIDOR	16
ILUSTRACIÓN 14: VIRTUALIZACIÓN DE SISTEMAS OPERATIVOS	17
ILUSTRACIÓN 15: VIRTUALIZACIÓN DE LAS FUNCIONES DE RED	18
ILUSTRACIÓN 16: INSTALANDO KVM	21

## Objetivos de la virtualización

### General

Comprender las diferentes formas de interactuar con el ambiente tecnológico, a medida que vaya creciendo nuestro conocimiento así ira creciendo el desarrollo de las tecnologías de este modo se desarrollara nuevas formas de virtualizar diferentes tipos de sistemas u operaciones las cuales nosotros mismo podríamos realizar.

### Específicos

- Aprovechar de forma más eficiente los recursos físicos disponibles.
- Facilitar el trabajo a los administradores de sistemas.
- Dar mejor servicio al usuario en general gracias a que la gestión de la infraestructura es más rápida y eficiente.
- Aumentar la eficacia del hardware y el software, y simplificar el mantenimiento del sistema mediante el uso exclusivo de entornos virtuales.
- Ejecutar aplicaciones con diferentes programas de actualización o SRU en el mismo servidor mediante zonas de núcleo Oracle Solaris.
- Mantener el aislamiento y supervisar la aplicación mediante la ejecución de cada aplicación en su propia zona o dominio invitado.

## Resumen

Los hipervisores son aplicaciones que presentan a los sistemas operativos virtualizados (sistemas invitados) una plataforma operativa virtual (hardware virtual), a la vez que ocultan a dicho sistema operativo virtualizado las características físicas reales del equipo sobre el que operan.

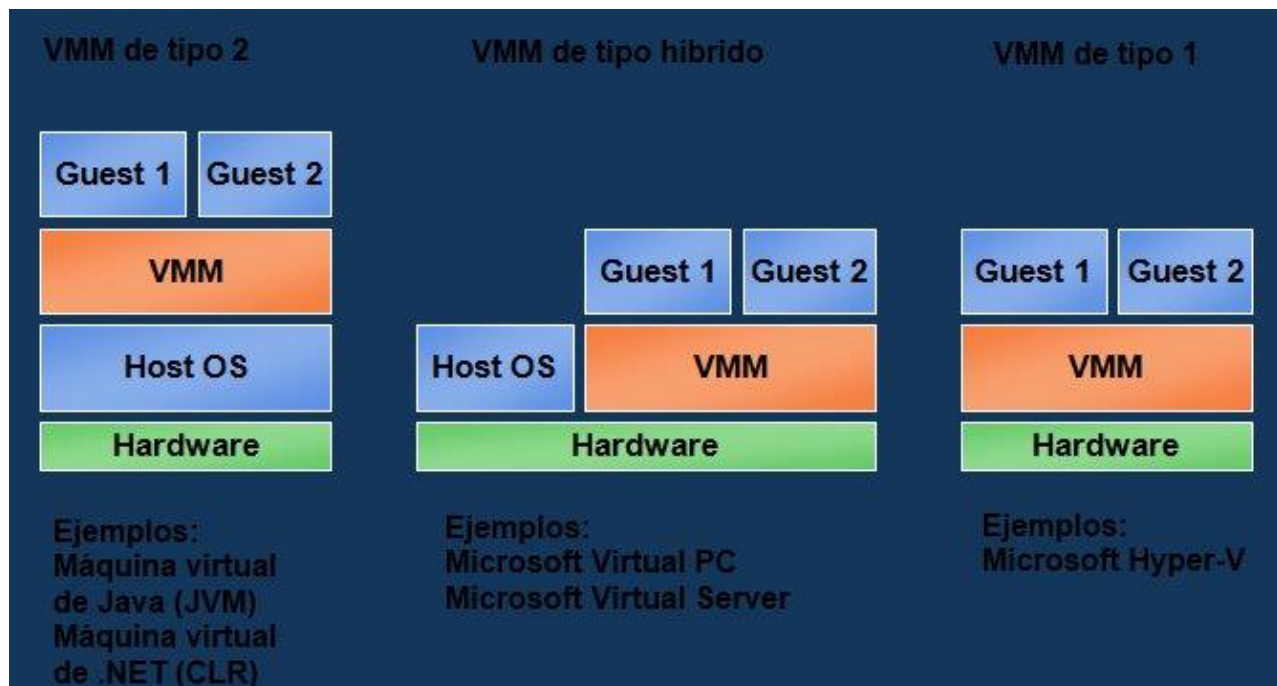
Los hipervisores también son los encargados de monitorizar la ejecución de los sistemas operativos invitados.

Con el uso de hipervisores es posible conseguir que múltiples sistemas operativos compitan por el acceso simultáneo a los recursos hardware de una máquina virtual de manera eficaz y sin conflictos.

Nota: es muy frecuente que a los hipervisores en general se les aplique el término VMM (Monitores de máquina virtual), mientras que el término “Hypervisor” se reserva para los hipervisores de tipo 1.

Nota: en algunas clasificaciones es posible que se incluyan los VMM de tipo híbrido dentro de los de tipo 2.

ILUSTRACIÓN 2: TIPOS DE HYPERVISOR



## Palabras claves

ILUSTRACIÓN 3: PALABRAS CLAVES



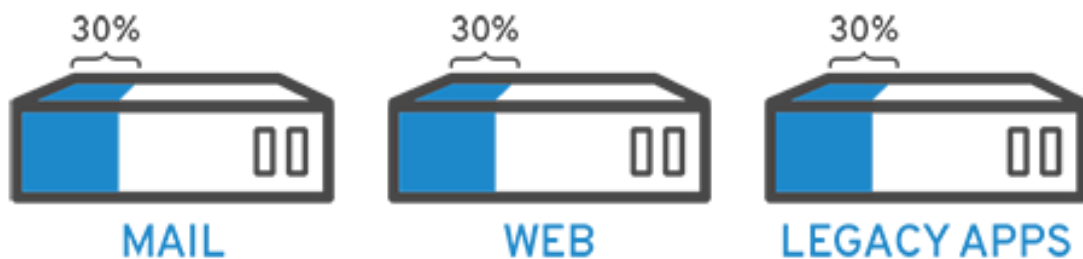
[Hypervisor](#), [Máquina virtual](#), [KVM](#), [Xen](#), [QEMU](#), [Libvirt](#), [CitrixXenServer](#), [Virtualización](#), [VMWare](#), [VirtualBox](#), [OpenVZ](#), [LXC](#), [Hyper-V](#), [IBM PowerVM](#).

## Introducción

La virtualización es una tecnología que permite crear servicios de TI útiles, con recursos que están unidos tradicionalmente al hardware. Permite utilizar toda la capacidad de una máquina virtual, porque distribuye sus capacidades en varios usuarios o entornos.

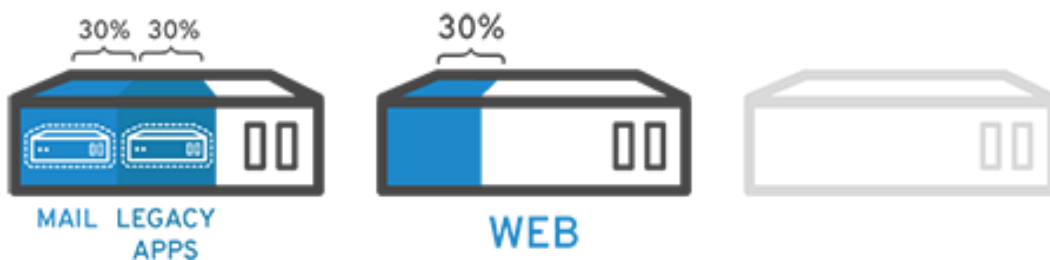
En términos más prácticos, imagine que tiene tres servidores físicos con propósitos dedicados individuales. Uno es un servidor de correo, otro es un servidor web y el otro ejecuta aplicaciones heredadas internas. Se utiliza cerca del 30% de la capacidad de cada servidor, es decir, solo una parte de su potencial de ejecución. Pero como las aplicaciones heredadas siguen siendo importantes para sus operaciones internas, tiene que conservarlas junto con el tercer servidor que las hospeda, ¿cierto?

ILUSTRACIÓN 4: SERVIDORES 1



En términos "tradicionales", la respuesta es sí. Generalmente, era más fácil y confiable ejecutar tareas individuales en servidores individuales: un servidor, un sistema operativo y una tarea. No era sencillo asignar múltiples tareas a un servidor. Pero la virtualización permite dividir el servidor de correo en dos servidores únicos que pueden administrar tareas independientes para que las aplicaciones heredadas se puedan migrar. Se utiliza el mismo hardware, pero de manera más eficiente.

ILUSTRACIÓN 5: SERVIDORES 2



Si pensamos en la seguridad, es posible volver a dividir el primer servidor para que administre otra tarea, y aumentar su uso de un 30% a un 60%, y luego a un 90%. Una vez que lo logra, los servidores que quedan vacíos se pueden reutilizar para otras tareas o retirarse todos juntos para reducir los costos de refrigeración y mantenimiento.



**libvirt** (The virtualization API) es un conjunto de herramientas para interactuar con las capacidades de virtualización de las versiones recientes de Linux (y otros sistemas operativos), liberado bajo la licencia "GNU Lesser General Public License".

libvirt soporta un gran conjunto de tecnologías entre las que se destacan:

- KVM/QEMU
- Xen (en Linux y Solaris)
- LXC
- OpenVZ
- El kernel paravirtualizado User Mode Linux
- VirtualBox
- VMware ESX y GSX
- VMware Workstation y Player
- Microsoft Hyper-V
- IBM PowerVM
- El hipervisor Parallels
- Redes virtuales usando bridging, NAT, VEPA y VN-LINK
- Almacenamiento en discos IDE/SCSI/USB, FibreChannel, LVM, iSCSI y NFS

libvirt permite administrar hosts de forma remota utilizando encriptación TLS y certificados x509; autenticación con Kerberos y SASL; control de acceso local mediante PolicyKit; Zeroconf utilizando Avahi; administración de máquinas virtuales, redes y almacenamiento; y una API cliente portable para Linux, Solaris y Windows.

Entre su amplia variedad de aplicaciones se destacan: virsh y virt-manager.

## Marco Teórico

ILUSTRACIÓN 6: MARCO TEÓRICO - MÁQUINAS VIRTUALES



Aunque hay muchos rasgos que se podrían destacar como candidatos para formar parte de una definición del concepto de máquina virtual (a partir de ahora MV), una MV se puede definir sencillamente como una capa de abstracción que separa el funcionamiento de un ordenador de su hardware.

Las MV se construyeron para simplificar el proceso del control del hardware de un ordenador porque extienden y enmascaran la funcionalidad del hardware a través de procedimientos y datos abstractos. Se pueden identificar dos tipos de MV concretas: las que juegan un papel en la preparación de un programa para su ejecución (tiempo de compilación) y las que permiten la ejecución de dicho programa.

Las Máquinas Virtuales constituyen un medio para proporcionar a un ambiente determinado de un sistema operativo, la posibilidad de ejecutar software que está disponible en otra plataforma esta plataforma es otro sistema operativo. Definimos a una Máquina Virtual como "un sistema de software que ofrece una copia exacta del hardware sobre el cual, la Máquina Virtual, se está ejecutando".

Este enfoque de Máquina Virtual permite que se ejecuten, de manera concurrente, varios Sistemas Operativos en una misma plataforma de hardware. La parte central de una Máquina Virtual es el Monitor de Máquina Virtual. Los Monitores de Máquinas Virtuales son programas que dan el soporte para la ejecución de las Máquinas Virtuales sobre alguna plataforma determinada.

## MÁQUINAS VIRTUALES DE SISTEMA

Las máquinas virtuales de sistema, también llamadas máquinas virtuales de hardware, permiten a la máquina física subyacente multiplexarse entre varias máquinas virtuales, cada una ejecutando su propio sistema operativo. A la capa de software que permite la virtualización se la llama monitor de máquina virtual o "hypervisor".

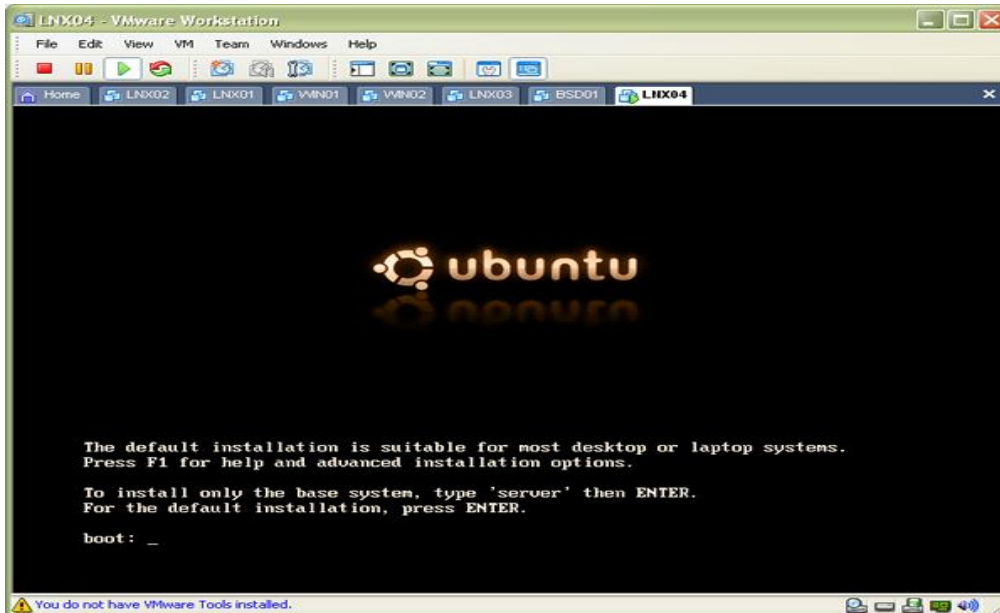
ILUSTRACIÓN 7: MÁQUINA VIRTUAL DE SISTEMA



## MÁQUINAS VIRTUALES DE PROCESO

Una máquina virtual de proceso, a veces llamada "máquina virtual de aplicación", se ejecuta como un proceso normal dentro de un sistema operativo y soporta un solo proceso. La máquina se inicia automáticamente cuando se lanza el proceso que se desea ejecutar y se para cuando éste finaliza. Su objetivo es el de proporcionar un entorno de ejecución independiente de la plataforma de hardware y del sistema operativo, que oculte los detalles de la plataforma subyacente y permita que un programa se ejecute siempre de la misma forma sobre cualquier plataforma.

ILUSTRACIÓN 8: MÁQUINA VIRTUAL DE PROCESO



Dentro del concepto que tenemos de las máquinas virtuales, existen dos grandes programas que permiten la emulación de los sistemas operativos sin afectar el equipo físico sobre el cual se está trabajando:

**VirtualBox** es un software de virtualización para arquitecturas x86 que fue desarrollado originalmente por la empresa alemana Innotek GmbH, pero que pasó a ser propiedad de la empresa Sun Microsystems en febrero de 2008 cuando ésta compró a innotek. Por medio de esta aplicación es posible instalar sistemas operativos adicionales, conocidos como “sistemas invitados”, dentro de otro sistema operativo “anfitrión”, cada uno con su propio ambiente virtual. La versión que desarrollamos en el trabajo es la última que sacó Sun Microsystem para desarrollar.

## Materiales y Métodos (Metodología)

ILUSTRACIÓN 9: HYPERVISOR KVM



KVM

Se inició como un proyecto centrado en una solución VDI para clientes Windows y GNU/Linux. Aunque la tecnología de virtualización data de la década de los sesenta, recién a principios del año 2000 se comenzó a adoptar más ampliamente. Las tecnologías que posibilitaron la virtualización, como los hipervisores, se desarrollaron hace muchas décadas para permitir que muchos usuarios accedieran simultáneamente a computadoras que realizaban procesamiento por lotes. El procesamiento por lotes era un tipo de informática popular en el sector comercial que ejecutaba tareas rutinarias miles de veces y muy rápidamente (como las nóminas).

Sin embargo, durante las décadas posteriores, surgieron otras soluciones que respondían al problema de tener una gran cantidad de usuarios y una sola máquina, pero, aun así, la virtualización no siguió los mismos pasos. Una de esas soluciones era el tiempo compartido, el cual aislaba a los usuarios en sistemas operativos, que inadvertidamente originó otros sistemas operativos, como UNIX, que finalmente dio paso al surgimiento de Linux®. Mientras tanto, la virtualización siguió sin adoptarse ampliamente y se mantuvo como una tecnología de nicho.

Ahora bien, avancemos a la década de los noventa. La mayoría de las empresas tenían servidores físicos y pilas de TI de un solo proveedor, lo cual no permitía que las aplicaciones heredadas se ejecutaran en un hardware de otro proveedor. A medida que las empresas actualizaban sus entornos de TI con servidores básicos, sistemas operativos y aplicaciones menos costosos y de diferentes proveedores, el hardware físico se usaba de manera insuficiente y cada servidor podía ejecutar solo una tarea específica del proveedor.

En este momento, la virtualización hizo un verdadero despegue. Resultó ser la solución natural para dos problemas: las empresas podían dividir los servidores y ejecutar aplicaciones heredadas en varios tipos y versiones de sistemas operativos. Los servidores se empezaron a utilizar más eficientemente (o quedaron directamente en desuso) y, en consecuencia, se redujeron los costos relacionados con las compras, la instalación, la refrigeración y el mantenimiento.

La aplicación generalizada de la virtualización ayudó a reducir la dependencia de un solo proveedor y constituyó la base del cloud computing. Actualmente, su uso se ha expandido tanto en las empresas que a menudo se necesita un software de administración de virtualización especializado para realizar el seguimiento.

KVM es un hipervisor tipo 1, que proporciona una solución de virtualización completa para Sistemas Operativos Linux en arquitecturas que cuenten con extensiones de virtualización. A lo largo de este curso trataremos con detalle KVM, QEMU y Libvirt; explicaremos cómo estos componentes de código abierto, se relacionan e interactúan entre sí para ensamblar y proveer una infraestructura de TI virtualizada.

Analizaremos con nivel de detalle la implementación del hipervisor KVM, asegurando la infraestructura virtual y la administración de recursos de virtualización. Realizaremos laboratorios que demuestren el despliegue, creación y mantenimiento de VM, integraciones con elementos de infraestructura de red, y dispositivos.

Exploraremos casos prácticos que se presentan habitualmente en las organizaciones, detallando la manera en la que pueden ser atendidos, permitiendo a las personas que tomen el curso, adquirir una perspectiva en la implementación y administración de una solución de virtualización a través de KVM.

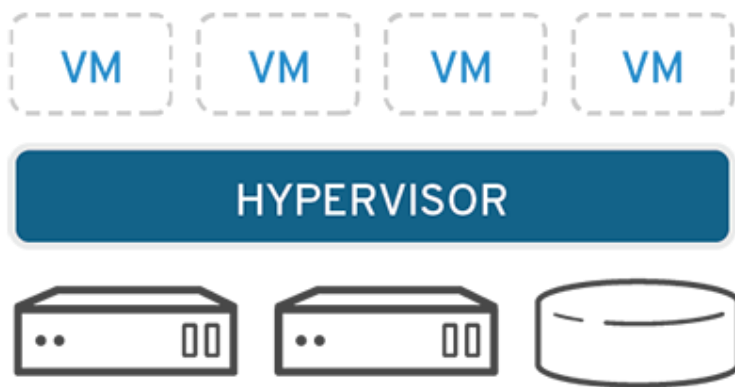
Este material está dirigido a todo aquel interesado en los hipervisores, interesados en la construcción de una infraestructura de virtualización confiable, eficiente y de código abierto; administradores de sistemas que deseen ampliar sus conocimientos para cubrir tópicos de arquitecturas o entornos de TI virtual avanzados.

### ¿Cómo funciona KVM?

KVM convierte a Linux en un hipervisor de tipo 1 (sin sistema operativo). Todos los hipervisores necesitan algunos componentes al nivel del sistema operativo (por ejemplo, administrador de memoria, planificador de procesos, pila de entrada o salida (E/S), controladores de dispositivos, gestión de seguridad, pila de red y más) para ejecutar las máquinas virtuales. KVM cuenta con todos estos componentes porque es parte del kernel de Linux. Cada máquina virtual se implementa como un proceso regular de Linux, programada por el planificador estándar de Linux con hardware virtual dedicado como tarjeta de red, adaptador de gráficos, CPU, memoria y discos.

El software denominado hipervisores separa los recursos físicos de los entornos virtuales, es decir, todo lo que necesitan los recursos. Los hipervisores pueden conformarse como elementos principales de un sistema operativo (como una computadora portátil) o se pueden instalar directamente en el hardware (como un servidor), que es la forma en que la mayoría de las empresas virtualizan. Los hipervisores toman los recursos físicos y los dividen de manera tal que los entornos virtuales puedan usarlos.

ILUSTRACIÓN 10: KVM



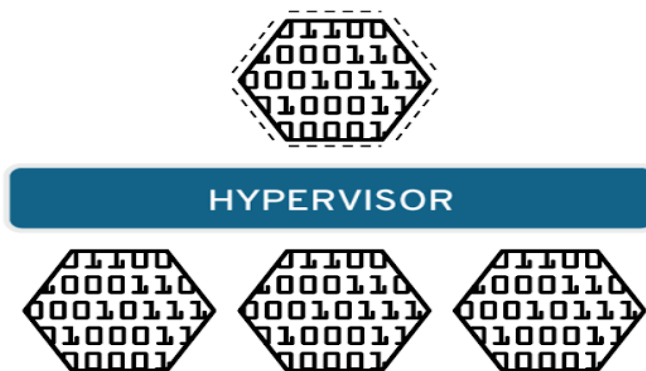
Los recursos se dividen según las necesidades, desde el entorno físico hasta los numerosos entornos virtuales. Los usuarios interactúan con los cálculos y los ejecutan dentro del entorno virtual (generalmente denominado máquina de guest o máquina virtual). La máquina virtual funciona como un archivo de datos único. Al igual que cualquier archivo digital, se puede migrar de una computadora a otra, abrir en cualquier computadora y prever que funcione de la misma manera.

Cuando el entorno virtual se está ejecutando y un usuario o programa emite una instrucción que requiere recursos adicionales del entorno físico, el hipervisor transmite la solicitud al sistema físico y guarda los cambios en la caché. Todo esto sucede a una velocidad prácticamente nativa (en especial, si la solicitud se envía a través de un hipervisor de open source basado en la máquina virtual basada en el kernel).

## Tipos de virtualización

### Virtualización de los datos

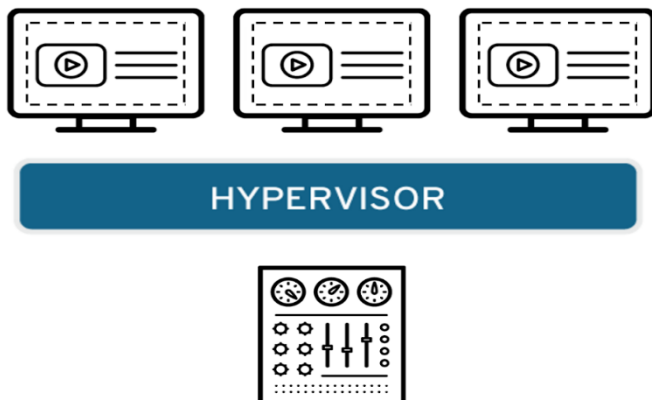
ILUSTRACIÓN 11: VIRTUALIZACIÓN DE DATOS



Los datos que se encuentran repartidos por todas partes se pueden consolidar en una fuente única. La virtualización de datos permite a las empresas tratar a los datos como si fueran una cadena de suministro; de esta manera, se obtienen las capacidades de procesamiento que pueden reunir los datos procedentes de varias fuentes, se acomodan con facilidad nuevas fuentes de datos y se transforman los datos de acuerdo con las necesidades de los usuarios. Las herramientas de virtualización de datos (como Red Hat® JBoss® Data Virtualization) se enfrentan a múltiples fuentes de datos y permiten tratarlas como una única fuente, por lo que ofrece los datos necesarios, de la forma requerida, en el momento justo para cualquier aplicación o usuario.

## Virtualización de escritorios

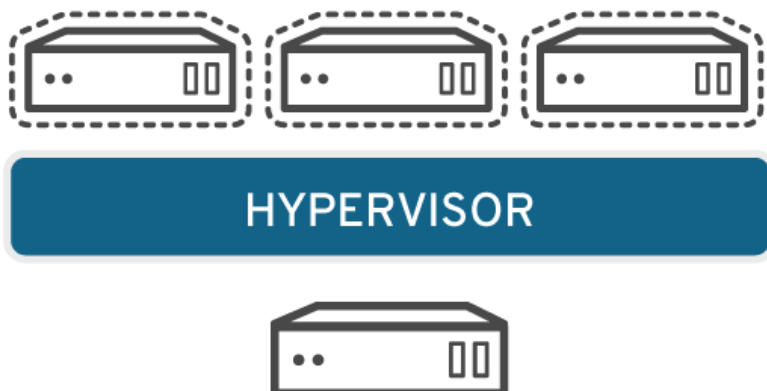
ILUSTRACIÓN 12: VIRTUALIZACIÓN DE ESCRITORIOS



La virtualización de escritorios (que se confunde fácilmente con la virtualización de los sistemas operativos que permite implementar múltiples sistemas operativos en una sola máquina) permite que un administrador central, o una herramienta de administración automatizada, implemente entornos simulados de escritorio en cientos de máquinas físicas de una sola vez. A diferencia de los entornos de escritorio tradicionales que se instalan, configuran y actualizan físicamente en cada máquina, la virtualización de escritorios permite que los administradores realicen configuraciones, actualizaciones y controles de seguridad de forma masiva en todos los escritorios virtuales.

## Virtualización de servidor

ILUSTRACIÓN 13: VIRTUALIZACIÓN DE SERVIDOR

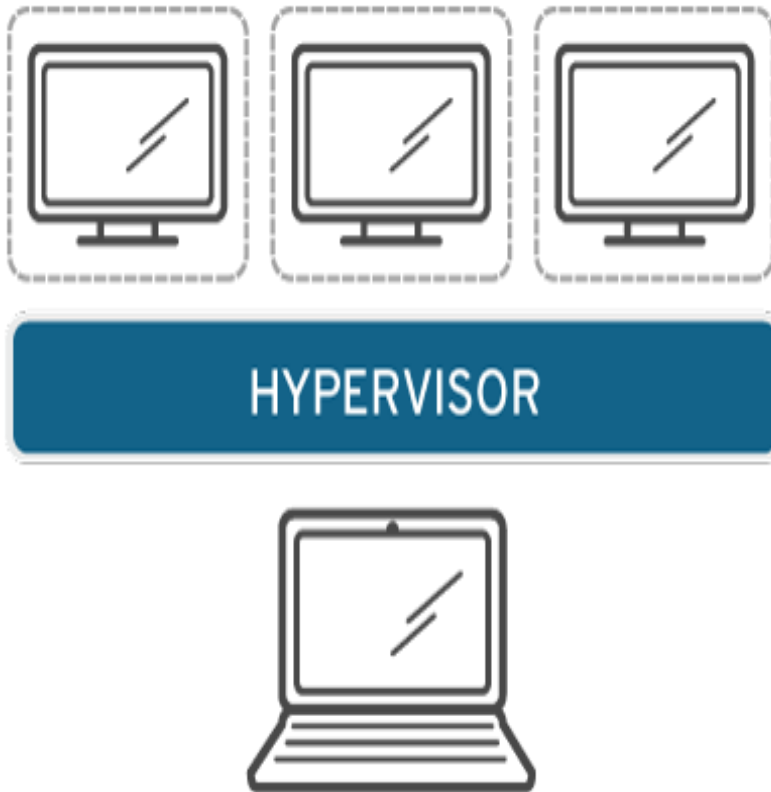


Los servidores son computadoras diseñadas para procesar un gran volumen de tareas específicas de forma muy efectiva para que otras computadoras (computadoras portátiles o de escritorio) puedan ejecutar otras tareas. Virtualizar un servidor le permite ejecutar más funciones específicas e implica dividirlo para que los componentes se puedan utilizar para realizar varias funciones.



## Virtualización del sistema operativo

ILUSTRACIÓN 14: VIRTUALIZACIÓN DE SISTEMAS OPERATIVOS

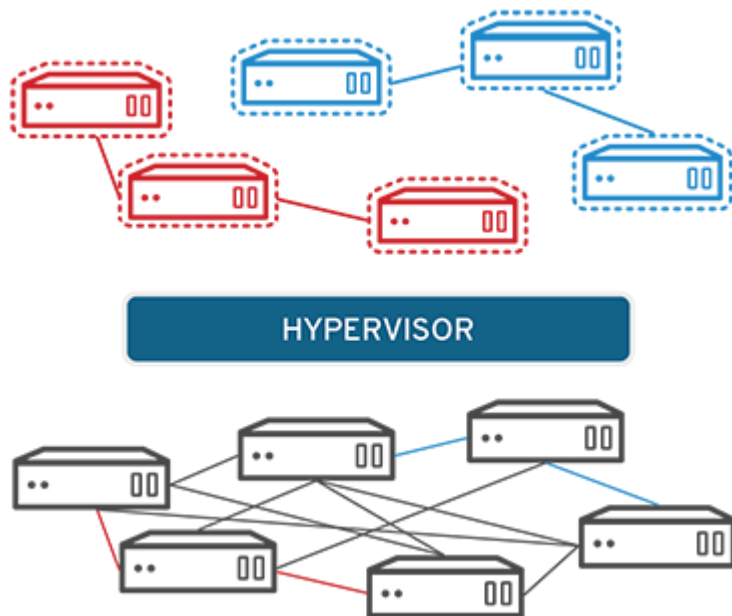


La virtualización del sistema operativo se realiza en el kernel, es decir, los administradores de tareas centrales de los sistemas operativos. Es una forma útil de ejecutar los entornos Linux y Windows de manera paralela. Las empresas también pueden insertar sistemas operativos virtuales en las computadoras, lo cual:

- Reduce el costo del hardware en masa, ya que las computadoras no requieren capacidades tan inmediatas.
- Aumenta la seguridad porque todas las instancias virtuales se pueden supervisar y aislar.
- Limita el tiempo que se destina a los servicios de TI, como las actualizaciones de software.

## Virtualización de las funciones de red

ILUSTRACIÓN 15: VIRTUALIZACIÓN DE LAS FUNCIONES DE RED



La virtualización de las funciones de red (NFV) separa las funciones clave de una red (como los servicios de directorio, el uso compartido de archivos y la configuración de IP) para distribuirlas en los entornos. Cuando las funciones del software se independizan de las máquinas virtuales en las cuales alguna vez estuvieron, las funciones específicas se pueden empaquetar en una nueva red y asignarse a un entorno. La virtualización de redes reduce la cantidad de componentes físicos (como conmutadores, routers, servidores, cables y centrales) que se necesitan para crear varias redes independientes y es particularmente conocida en el sector de las telecomunicaciones.

### Funciones de KVM

En pocas palabras, KVM es Linux: KVM es parte de Linux. Linux es parte de KVM. Todo lo que tiene Linux, KVM también lo tiene. Pero hay funciones específicas que hacen que KVM sea el hipervisor empresarial preferido.

### Seguridad

KVM utiliza una combinación de security-enhanced Linux (SELinux) y virtualización segura (sVirt) para la seguridad y el aislamiento mejorados de máquina virtual. SELinux establece los límites de seguridad para las máquinas virtuales. sVirt amplía las capacidades de SELinux, lo que permite que la seguridad de control de acceso obligatorio (MAC) se aplique a las máquinas virtuales huéspedes y evita los errores manuales de etiquetado.

### Almacenamiento

KVM puede usar cualquier almacenamiento compatible con Linux, incluidos algunos discos locales y el almacenamiento conectado a la red (NAS). La entrada/salida de rutas múltiples mejora el

almacenamiento y proporciona redundancia. KVM también es compatible con sistemas de archivos compartidos de tal manera que las imágenes de las máquinas virtuales se puedan compartir entre varios hosts. Las imágenes de disco dan soporte al aprovisionamiento ligero para asignar almacenamiento a pedido en lugar de desde el inicio.

### **Compatibilidad con el hardware**

KVM puede usar una amplia variedad de plataformas de hardware certificado y compatible con Linux. Debido a que los proveedores de hardware contribuyen periódicamente al desarrollo del kernel, las funciones de hardware más recientes a menudo se adoptan rápidamente en el kernel de Linux.

### **Gestión de memoria**

KVM hereda las funciones de gestión de memoria de Linux, incluidos el acceso a la memoria no uniforme y la fusión de la misma página del kernel. La memoria de una máquina virtual se puede intercambiar, tiene el soporte de grandes volúmenes para un mejor rendimiento, y se puede compartir o respaldar por un archivo de disco.

### **Migración en vivo**

KVM permite la migración en vivo, esto es la capacidad de mover una máquina virtual en funcionamiento de un host físico a otro sin que se interrumpa el servicio. La máquina virtual permanece encendida, las conexiones de red permanecen activas y las aplicaciones continúan ejecutándose mientras se reubica la máquina virtual. KVM también guarda el estado actual de la máquina virtual para que se pueda almacenar y retomar posteriormente.

### **Rendimiento y escalabilidad**

KVM hereda el rendimiento de Linux, ya que puede escalar hasta satisfacer la carga requerida si la cantidad de máquinas huéspedes y las demandas crece. KVM permite que las cargas de trabajo de las aplicaciones más exigentes se virtualicen y es la base para muchas configuraciones de virtualización empresarial, como los centros de datos y las nubes privadas (mediante OpenStack®).

### **Programación y control de recursos**

En el modelo de KVM, una máquina virtual es un proceso de Linux, programado y controlado por el kernel. El planificador de Linux permite el control detallado de los recursos asignados a un proceso de Linux y garantiza la calidad del servicio para un proceso en particular. En KVM, esto incluye el planificador completo, los grupos de control, los espacios de nombre de red y las extensiones en tiempo real.

### **Latencia más baja y mayor priorización**

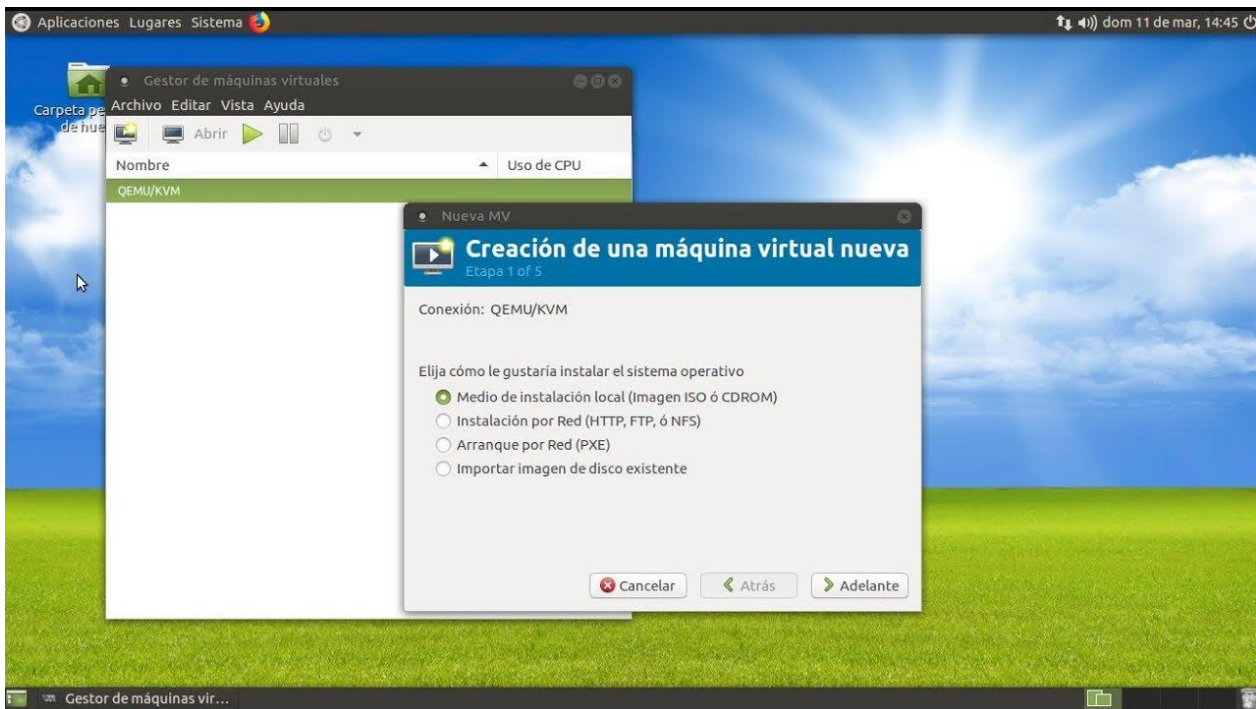
El kernel de Linux cuenta con extensiones en tiempo real que permiten que las aplicaciones basadas en máquinas virtuales se ejecuten con una latencia más baja y una mejor priorización (frente a sin sistema operativo). El kernel también divide los procesos que requieren largos períodos de computación en componentes más pequeños, que luego se programan y se procesan en consecuencia.

## **Gestionar KVM**

Es posible gestionar manualmente un grupo de máquinas virtuales (VM) que se encienden en una sola estación de trabajo sin una herramienta de gestión. Las empresas grandes usan software de gestión de virtualización que sirve como interfaz con los entornos virtuales y el hardware físico subyacente para simplificar la gestión de recursos, mejorar el análisis de datos y optimizar las operaciones. Red Hat creó Red Hat Virtualization exactamente con este propósito.

## Instalación de KVM

ILUSTRACIÓN 16: INSTALANDO KVM



Por lo tanto, estamos hablando de una solución de virtualización para utilizar en los sistemas Linux. KVM utiliza Virt-Manager como administrador de máquinas virtuales y un Qemu modificado como hypervisor. Yo he probado la instalación en Linux Mint 18.3 y ha funcionado correctamente. Para instalarlo lo primero que debemos saber es si nuestro procesador es compatible. Para ello introduciremos los siguientes comandos:

```
$ cat /proc/cpuinfo | grep vmx # para CPUs Intel  
$ cat /proc/cpuinfo | grep svm # para CPUs AMD
```

Si el sistema no nos devuelve nada me temo que no tenemos soporte de virtualización en nuestro procesador o no lo tenemos activado. En caso afirmativo, instalamos los paquetes necesarios:

```
$ sudo apt-get install kvm libvirt-bin ubuntu-vm-builder bridge-utils
```

Nota: Si al ejecutar este comando nos manda un error de dependencias entonces hacer lo siguiente.

```
Sudo apt-get update
```

```
sudo apt-get install kvm libvirt-bin ubuntu-vm-builder bridge-utils
```

luego de haber hecho la instalacion ahora vamos a incluir el usuario que sería nuestro sistema en el grupo de kvm y para ello ejecutamos:

```
$ sudo adduser $USER kvm
```

Una vez realizado todos estos pasos tendremos que reiniciar la sesión, dependiendo de la versión de GNU/Linux que estemos utilizando. Si todo ha ido bien, tendremos una nueva entrada en nuestro menú Aplicaciones/Herramientas del Sistema/Administrador de Máquina Virtual desde donde podemos **comenzar la instalación de nuestras máquinas virtuales.**

## CUADRO COMPARATIVO ENTRE HERRAMIENTAS DE VIRTUALIZACIÓN

TABLA 1: CUADRO COMPARATIVO

Herramienta	Precio	Licencia	Arquitectura	Rendimiento	Sistema Operativo	Notas
<b>XEN</b>	Free/Depende el tiempo de uso y la versión. Stándar-edition 1 año: \$763.00	GPL	x86-x64	Virtualización rápida, virtualización media	GNU/Linux	Full virtualización necesita VT/AMD-V
<b>KVM</b>	Free/Depende el tiempo de uso	GPL/LGPL	x86-x64	Virtualización rápida, virtualización media	GNU/Linux	Full virtualización y para virtualización necesita VT/AMD-V
<b>VIRTUALBOX</b>	--	Privativa / GPLv2 (Opcionalmente CDDL para la mayoría de archivos en la fuente)	x86-x64	Virtualización Rápido / Muy rápido	Multiplataforma	Módulo de kernel GPL, RPD soporte USB propietario
<b>VMWARE ESXI</b>	€68 \$ 82.94	Software propietario	x86-x64	Virtualización rápida	GNU/Linux y Windows	Necesita módulos propietarios para el kernel

## Características relevantes: Virtualizadores más importantes

### XEN

#### Ventajas

- Desempeño casi nativo.
- Permite virtualización completa y para virtualización.
- Puede funcionar aun en hardware que no soporta virtualización completa.
- Permite la migración en caliente de los sistemas clientes.
- Existe la convivencia entre servidores virtualizados y servidores para virtualizados.

#### Desventajas

- Los clientes deben ser modificados para su funcionamiento.
- No es compatible con la interfaz avanzada de configuración de energía (ACPI, APM) en tecnologías portátiles.
- No todo el hardware esta soportado.
- Problemas con algunos drivers propietarios.
- No admite varios chips de plan y bridges cardbus.

### KVM

#### Ventajas

- Está incluida en el kernel a partir de la versión 2.6.20.
- Permite virtualización completa.
- Puede ejecutar huéspedes Linux de 32 y 64 bits, así como Windows de 32 bits.
- Utiliza QEMU para el manejo de las máquinas virtuales, funciona como un driver de virtualización.
- Los procesadores virtuales de una máquina virtual son simples hilos del proceso anfitrión.
- Los sistemas clientes no necesitan modificaciones.

#### Desventajas

- Necesita un procesador con soporte para Virtualization Technology.
- Falta de opciones directas para la ejecución de sistemas operativos de otras arquitecturas (Las versiones en desarrollo están desarrollando estas características).



## VMWARE

### Ventajas

- Solidez, estabilidad, seguridad.
- Admite drivers en los entornos emulados.
- indicado para consolidación de servidores e investigaciones técnicas.

### Desventajas

- Rendimiento mediocre del gestor de máquinas virtuales con hardware de bajo rendimiento.
- La versión gratuita es de uso personal y no empresarial.
- Al actualizar el kernel se debe ejecutar nuevamente el install

## VIRTUALBOX

### Ventajas

- Soporta el conjunto de instrucciones de virtualización en procesadores Intel VTx y AMD-V, esto mejora el rendimiento y lo sitúa con muy buena puntuación en recientes comparativas de virtualización como Virtualization in Linux.
- Mantiene una edición de código abierto Open Source Edition (OSE), aunque con unas pocas menos funcionalidades.
- Se puede instalar en una amplia variedad de sistemas operativos soportados: Debian, Fedora, Mandriva, Ubuntu, RedHat, Open Solaris, Mac OS X, Xandros, openSUSE, PCLinux OS
- Dispone de excelente documentación técnica y de usuario.
- Es fácil de instalar y poner en marcha en todas las plataformas.
- La interfaz de usuario está traducida a más de 20 idiomas.
- Acceso a la consola de cada máquina virtual por red vía Remote Display Protocol (RDP).
- Soporte de asignación de dispositivos USB 2.0 entre el anfitrión y la máquina virtual e incluso paso de USB sobre RDP. Esta capacidad es muy potente: podemos conectar dispositivos USB desde máquinas distintas al anfitrión utilizando el cliente RDP.

### Desventajas

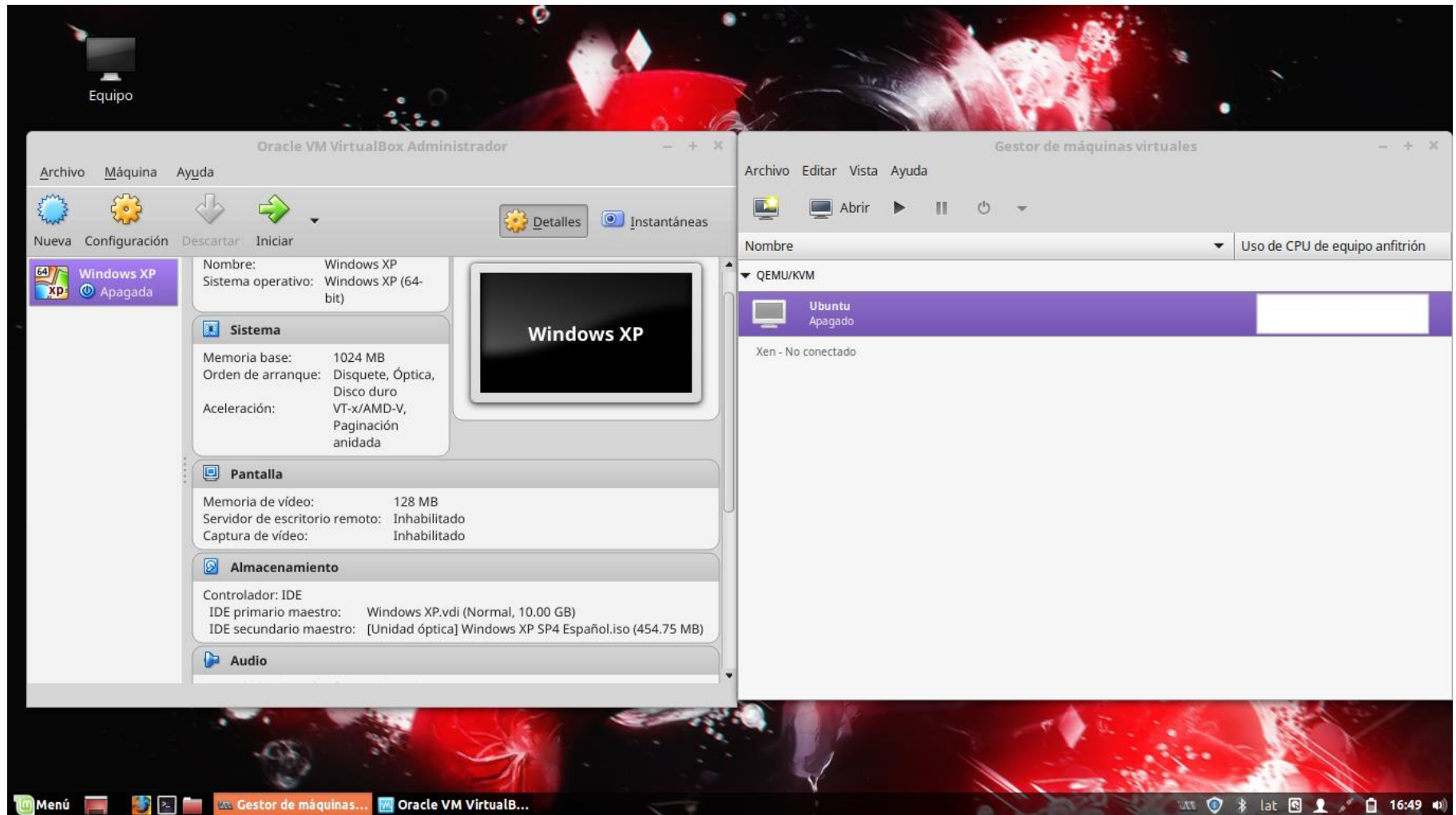
- Se requiere una computadora poderosa para poder emular una máquina virtual. Se requiere conocimiento técnico para poder realizar esta tarea.
- No se pueden utilizar instalaciones ya existentes de otros sistemas operativos.
- A veces tiende a ser difícil compartir la información de la máquina virtual con la máquina real.

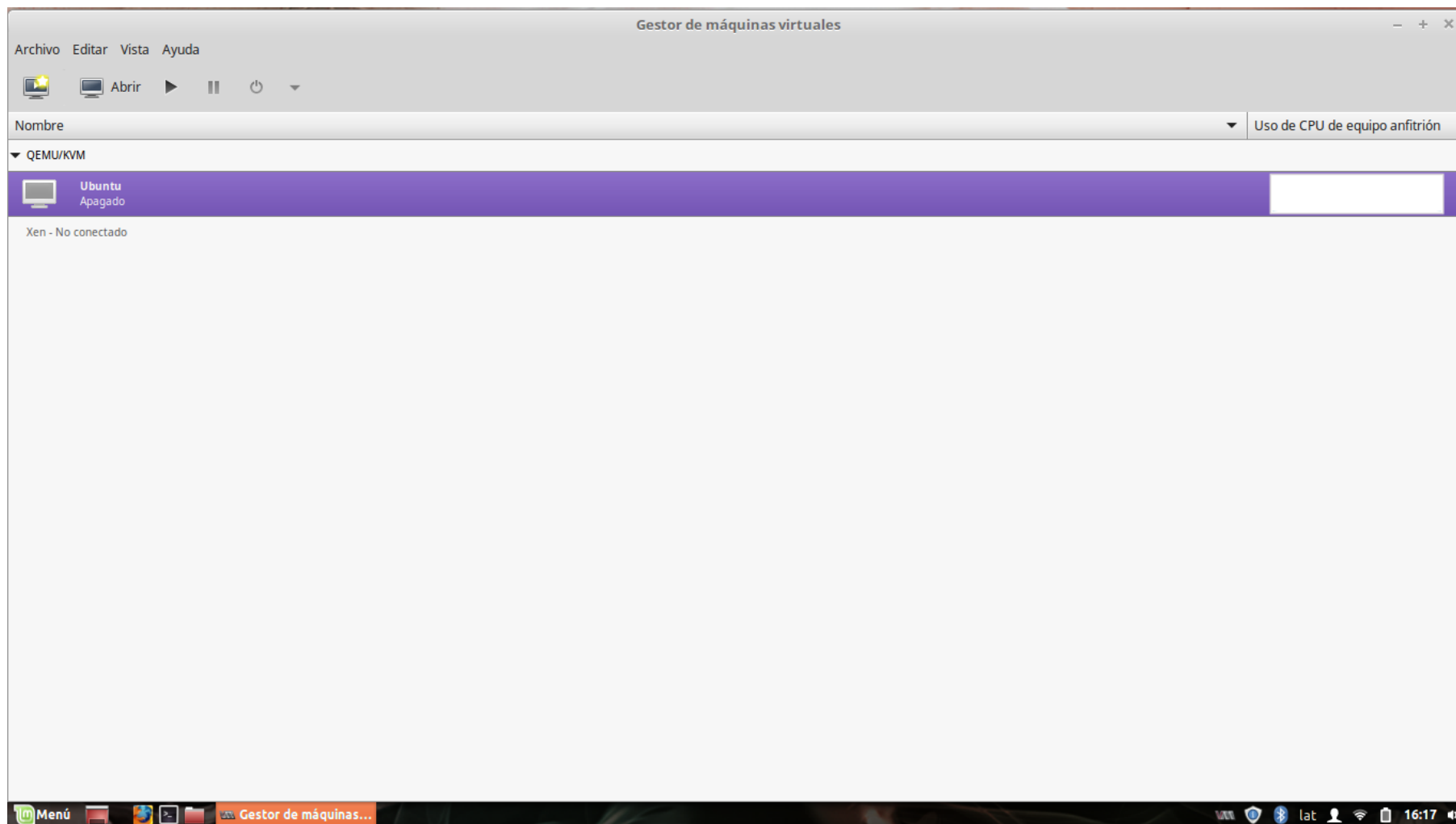
TABLA 2: PRESUPUESTO DEL PROYECTO

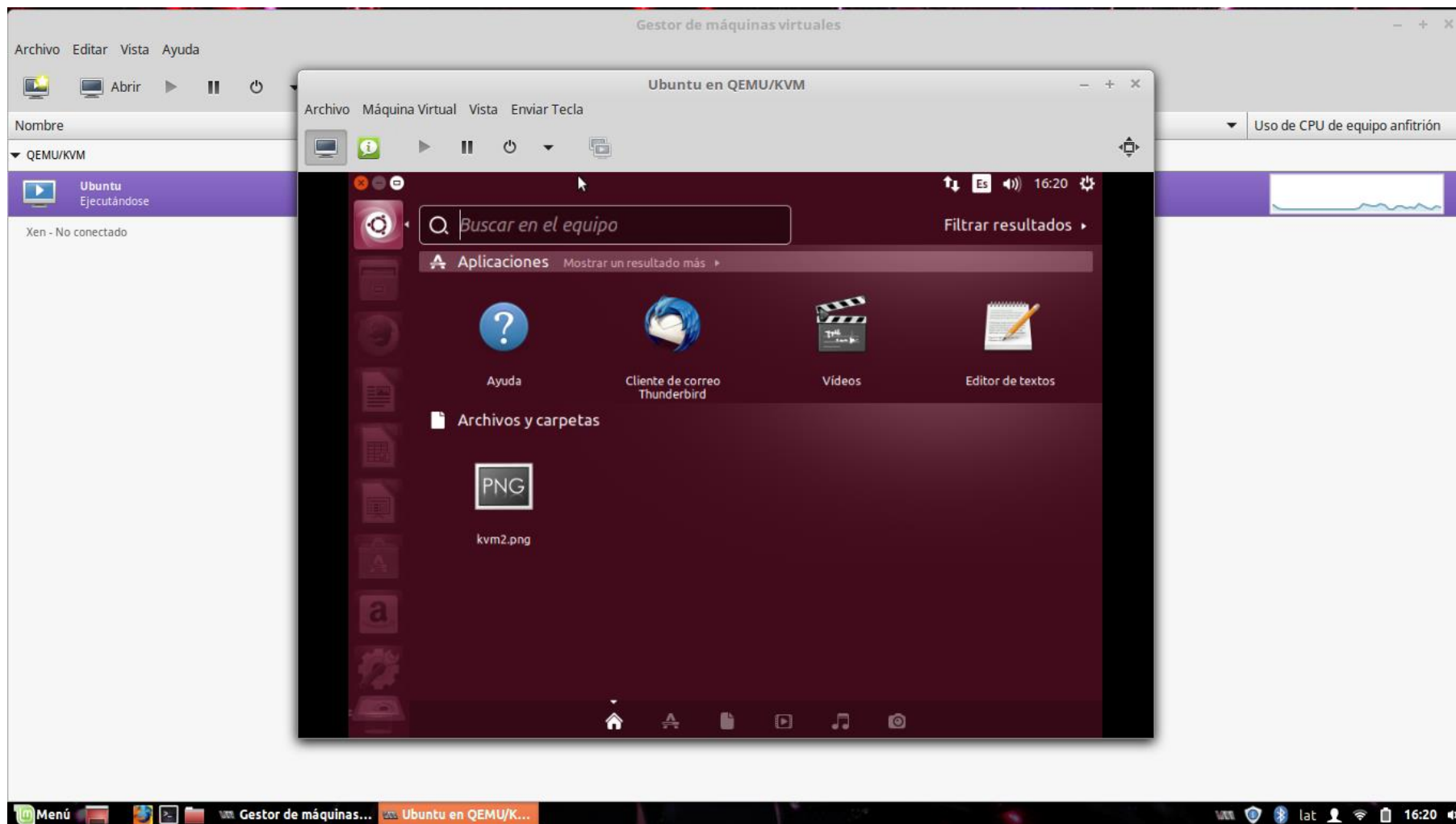
DENOMINACIÓN DE LA PARTIDA	UNIDAD DE MEDIDA	CANTIDAD	COSTO UNITARIO	COSTO TOTAL
<b>Personal</b>				
Asesor Metodológico	Mes	3	50.00	150.00
Asesor Temático	Mes	3	85.00	255.00
<b>Sub Total</b>				<b>\$ 405.00</b>
<b>Material de Escritorio</b>				
Papel Bond A4 80 gr.	Ciento	1	4.00	6.00
Lapiceros	Unidad	2	0.50	1.00
Resaltador	Unidad	2	3.50	7.00
Engrapadora	Unidad	1	3.50	7.00
<b>Sub Total</b>				<b>\$ 21.00</b>
<b>Procesamiento de Datos e Informática</b>				
Computadora	Equipo	2	250	500.00
Dispositivo USB	Unidad	2	25.00	50.00
<b>Sub Total</b>				<b>\$ 550.00</b>
<b>Servicio de Terceros</b>				
Internet	Hora	8	1.00	8.00
Impresión	Unidad	45	0.25	11.25
Fotocopiado	Unidad	90	0.20	18.00
Viáticos	Unitario	10	6.00	60.00
Honorarios	Finalizado	3	150.00	350.00
<b>Sub. Total</b>				<b>797.25</b>
<b>TOTAL:</b>				<b>\$ 1,773.25</b>

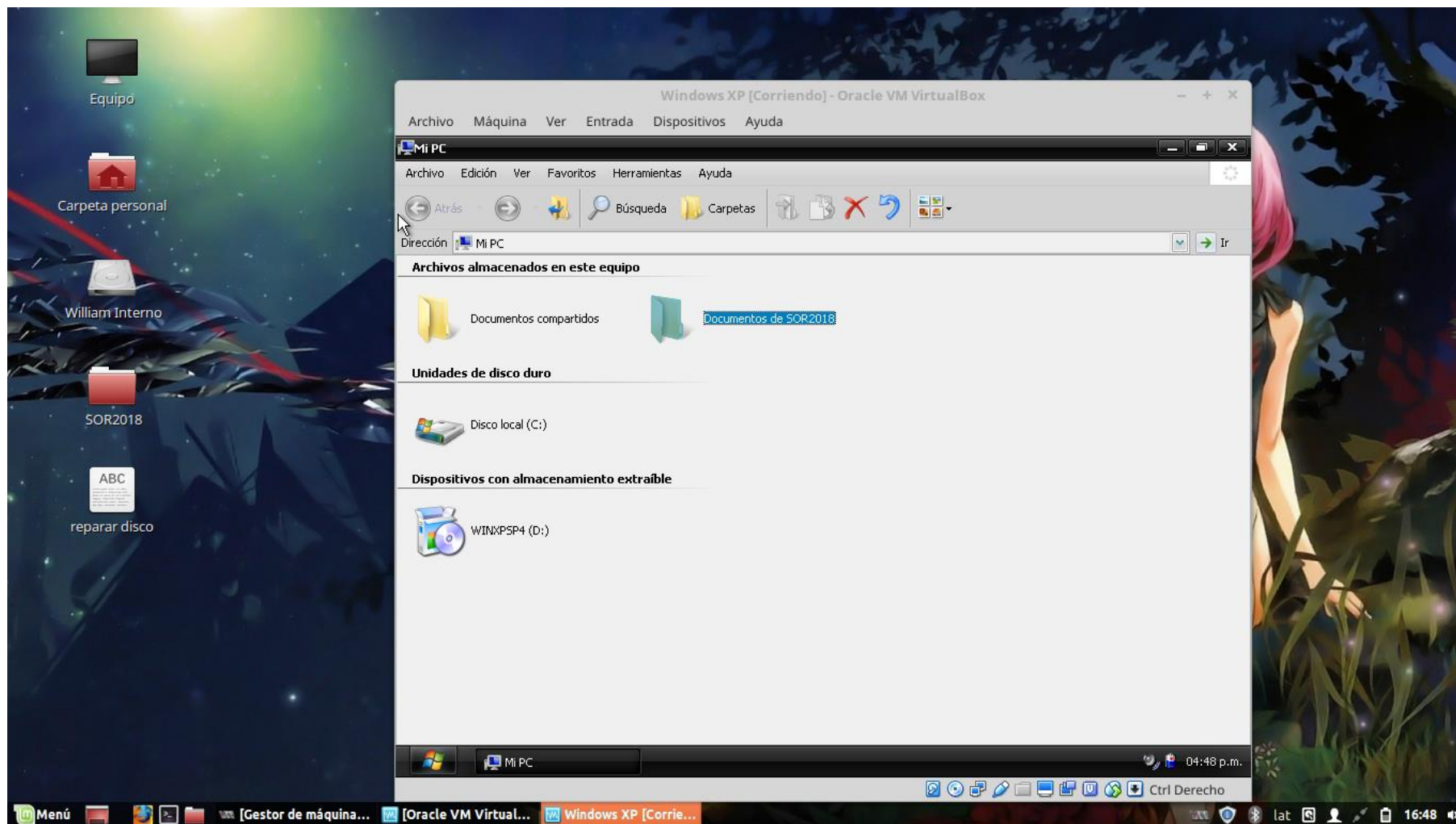
## Resultados

Basándonos en los bancos de pruebas detallados, se exponen los resultados con las siguientes capturas de pantallas obtenidas de una comparación entre KVM y VirtualBox usando GNU/Linux Mint 18.3









Tanto los sistemas virtualizados bajo Windows como bajo Linux físicos dieron los mismos resultados con el programa usado. Viendo los datos obtenidos podemos extraer las siguientes afirmaciones.

En lo que todos los programas identifican por igual es en el modelo de la CPU que por cierto es erróneo ya que nuestro Hardware de pruebas posee un Intel Core i3.

- Ninguno de los virtualizadores es capaz de obtener datos acerca del voltaje.
- La velocidad de reloj de todos los programas se encuentra en un rango similar.
- Se puede apreciar que en la detección del número de núcleos e hilos. No obstante hay que tener en cuenta la capacidad de nuestra maquina anfitriona ya que si le damos todos los nucleos esta podría bajar drásticamente su rendimiento.
- Las instrucciones SSE4 que hacen mejorar al núcleo de la CPU, sólo son detectadas por VMware.
- La cache de tercer nivel L3 del procesador tan sólo no es detectada en VirtualBox.
- Para VirtualBox detecta nuevamente una tarjeta gráfica propia del Software.

Esta prueba se ha realizado para extraer un poco de información con el objetivo de mostrar lo que puede ser detectado en un programa de virtualización. Aunque algunos datos no los muestre el virtualizador como por ejemplo la cache L3 de VirtualBox que no se muestra, no quiere decir que no exista.

## Conclusión

Debemos usar el software que más se adapte a nuestras necesidades, si las necesidades en virtualización son altas, virtualizar Server o maquinas Linux que necesitan potencia, VMWare sin duda sería la mejor opción. Si solo queremos virtualizar por trastear y conocer sistemas operativos distintos VirtualBox es una excelente herramienta, mientras que si virtualizamos solo sistemas operativos Windows para cualquier función básica utilizar Virtual PC. Con esto NO estamos diciendo que VirtualBox ni Virtual PC puedan correr maquinas servidor y demás, sino que es más adecuado correrlo en VMware, por fluidez, potencia y opciones.

En todo caso nosotros hemos decido optar por el Hypervisor VirtualBox para crear máquinas virtuales ya este tiene buen uso como también posee un rendimiento óptimo que las otras herramientas difícilmente contienen, dicho de otra manera, este sería una herramienta muy útil y muy fácil de usar sin embargo no descartamos KVM, el cual también es una buena herramienta, pero a diferencia de VirtualBox este tiene ciertas dificultades por lo cual mejor decidimos el Hypervisor VirtualBox.

XEN en la actualidad es ya un software suficientemente probado. Su licencia GPL lo hace muy atractivo para entornos de producción ya que abarata mucho los costes. Y combinado con sistemas de almacenamiento tipo RAID le aseguran una alta disponibilidad.

De este modo dejaremos para una próxima documentación la utilización de XEN, su configuración y gestión desde una interfaz gráfica como caso práctico en entornos GNU/Linux.



## Recomendaciones

1. Primero que nada, antes de elegir un sistema de virtualización debemos de saber para qué uso queremos darle. Si es para probar [distros](#) o simplemente para probar algunas aplicaciones sin tener la necesidad de usar virtualización por hardware, les recomiendo [VirtualBox](#). Su uso es muy sencillo, con tal sólo instalar el software, ya podrán instalar de una manera muy sencilla las máquinas virtuales (**guest**).
2. Si se va a virtualizar (**en serio**), o sea si se necesita usar realmente virtualización por hardware, recomendamos [KVM](#).
3. Si van a usar KVM asegúrense que en el **BIOS** este activada la opción de virtualizar por hardware, o sino les va a ser muy lento su uso.
4. Si van a acceder a sus Guest; utilicen:  
Por *ssh*  
Por *virt* (una gran aplicación por consola). Ejemplo: **sudo virt-viewer --connect qemu://proyectosbeta@192.168.0.193/system debian6Prueba**
5. Pueden verificar el consumo de CPU, memoria, etc. de sus máquinas virtuales (guest) con **virt-top**.
6. En todos los Guest instalen siempre **acpi**, **acpid** y **acpi-support**, porque en las versiones anteriores de KVM con Guest Ubuntu, al querer apagar la máquina, no se pagaba bien; entonces para no tener problemas con eso, les recomiendo que instalen siempre esos paquetes.

## Glosario

### A

**Administrador de E/S por bloques:** Este es el encargado de controlar el funcionamiento de los discos.

**Administrador de E/S por Bytes:** Son procedimientos de bajo nivel que se encargan de realizar la comunicación entre la computadora y aquellos periféricos que manejan bytes, como, por ejemplo, la impresora, la consola, entre otros.

### Anfitrión:

El sistema operativo del equipo anfitrión ejecuta huéspedes virtualizados.

### Aplicaciones:

El término "aplicación" es una forma abreviada para designar a un programa de aplicación. Un programa de aplicación es aquél diseñado para desempeñar una función específica directamente para el usuario o, en algunos casos, para otro programa de aplicación. Los ejemplos de estas aplicaciones incluyen los procesadores de textos, programas de bases de datos, navegadores de la Red, herramientas de desarrollo, programas de dibujo, pintura y edición de imágenes, y programas de comunicaciones. Las aplicaciones usan los servicios del sistema operativo del ordenador y otras aplicaciones de apoyo. La solicitud formal y los medios de comunicarse con otros programas que usa un programa de aplicación son lo que se denomina interfaz de programa de aplicación (application program interface, API).

### Armada:

Es un mecanismo de control de interrupciones.

### B

### Bit de Espera:

Es un bit dentro de la Pep.

### Bare-metal:

El término bare-metal (vacío) se refiere a la arquitectura física subyacente de una computadora. Ejecutar un sistema operativo en un bare-metal es otra forma de referirse al hecho de ejecutar una versión sin modificar de un sistema operativo en el hardware físico. Un ejemplo la ejecución de un sistema operativo sobre bare-metal, es el sistema operativo instalado normalmente.

### C

### Canales:

Procesadores dedicados.

### CMS:

(Sistema Supervisor de la Conversación) es un sistema de usuario único que ofrece potentes funciones para el desarrollo de programas, Creación del sitio web, Gestión y mantenimiento del sitio web, Administración del sitio web y del propio CMS.

Para hacernos una idea de la versatilidad y potencia de los gestores de contenidos, veamos a continuación algunos tipos de sitios web que pueden construirse con un CMS Web comercial, muchas veces sin necesidad de adaptaciones específicas.

- Sitio web corporativo
- Blog
- Plataforma de eCommerce
- Plataforma de Cursos Online

### Código:

O programa de máquina; es el conjunto de instrucciones máquina que tiene por objeto que el computador realice una determinada función.

### Controlador de dispositivo:

Software que controla un dispositivo que está conectado a, o es parte de, una computadora. (En inglés: device driver).

### **Controladores para-virtualizados:**

Los controladores para-virtualizados son controladores de dispositivo que operan en huéspedes de Linux completamente virtualizados. Estos controladores aumentan ampliamente el rendimiento de red y la E/S de dispositivo de bloque para huéspedes completamente virtualizados.

### **CPU virtualizado:**

Un sistema tiene una cantidad de CPU virtuales (VCPU) relativas al número de núcleos de procesador físico. El número de VCPU es finito y representa el número total de VCPU que se pueden asignar a máquinas virtuales huéspedes.

### **D**

#### **Demonio:**

Un demonio es un programa que funciona sin intervención humana, para cumplir una tarea determinada. Por ejemplo, lpd es un demonio que controla el flujo de los trabajos de impresión en una impresora

#### **Desvíos:**

Interrupciones debidas a errores de programación, producidas por el procesador cuando intenta ejecutar una instrucción.

#### **Dirección IP:**

Las direcciones IP son el método mediante el cual se identifican los ordenadores individuales (o, en una interpretación más estricta, las interfaces de red de dichos ordenadores) dentro de una red TCP/IP. Todas las direcciones IP consisten en cuatro números separados por puntos, donde cada número está entre 0 y 255.

#### **Dispositivos de E/S:**

Componente mecánico que se conecta a la Unidad Central de Procesamiento a través de las unidades de Entrada y Salida.

#### **Direcciones MAC:**

La dirección de Control de acceso de medios es la dirección de hardware para el controlador de interfaz de red. En el contexto

de virtualización las direcciones se deben generar para interfaces de red virtuales con cada MAC en su dominio local único.

#### **dispositivo phy:**

The phy device parameter allows guest's to access physical disks. Physical disks includes:

- Volúmenes LVM (por ejemplo, /dev/VolGroup00/LogVol02),
- particiones de disco (por ejemplo, /dev/sda5), y
- dispositivos de bloques completos (por ejemplo, /dev/sda).

El modo físico ofrece el mejor desempeño ya que el hipervisor sorteas las capas adicionales de software que se encuentren en el equipo, aunque con esto se pierde algo de flexibilidad a la hora de administrar el dispositivo.

### **E**

#### **E/S:**

Entrada / Salida.

#### **ENIAC:**

(electronic numerical integrator analyzer and computer), uno de los primeros computadores construídos.

#### **Estado problema:**

Modo Usuario; Este es el modo en el que se ejecutan los programas del usuario, en el cual sólo se pueden ejecutar instrucciones no privilegiadas.

#### **Estado Supervisor:**

Modo Núcleo; En este modo se pueden ejecutar todas las instrucciones privilegiadas y no privilegiadas, teniendo acceso total al hardware.

#### **Estado:**

El estado es un bit dentro de la PEP.

#### **Fusión SamePage del Kernel:**

El módulo de Fusión SamePage del Kernel (KSM, por las iniciales en inglés de Kernel SamePage Merging) es utilizado por el hipervisor KVM de modo de permitir a los huéspedes compartir páginas de memoria

idénticas. Las páginas compartidas son por lo general bibliotecas comunes, u otros tipos similares de datos de gran utilización. KSM puede incrementar el desempeño y la densidad ciertos huéspedes, manteniendo estas bibliotecas en caché disponibles para varios de ellos

## H

### **Hardware:**

Son todos los componentes físicos que componen un computador.

### **Hipervisor:**

Un **hipervisor** (en inglés *hypervisor*) o **monitor de máquina virtual** (*virtual machine monitor*) es una plataforma que permite aplicar diversas técnicas de control de virtualización para utilizar, al mismo tiempo, diferentes sistemas operativos (sin modificar o modificados, en el caso de paravirtualización) en una misma computadora. Es una extensión de un término anterior, «supervisor», que se aplicaba a los kernels de los sistemas operativos de computadora.

## I

### **Identificador único universal:**

Un Identificador universal único (UUID) es un método estándar de nomenclatura para dispositivos, sistemas y algunos objetos de software en entornos de informática distribuidos. Los tipos de UUID en virtualización incluyen:

identificadores de sistema de archivo ext2 y ext3, identificadores de dispositivos RAID, iSCSI e identificadores de dispositivo LUN, direcciones MAC e identificadores de máquinas virtuales.

**Instrucciones no privilegiadas:** Todas aquellas que su ejecución no afecta los recursos de la máquina de forma directa.

### **Instrucciones privilegiadas:**

Aquellas instrucciones cuya ejecución afecta directamente los recursos de la máquina.

**Interrupción Asíncronica:** Son aquellas interrupciones que ocurren de forma simultánea con el procesamiento y son generadas por un agente distinto al procesador.

### **Interrupciones:**

Una interrupción es un llamado forzoso a un procedimiento.

## J

### **JCL:**

(Job Control Language), es el lenguaje de control de trabajo que permite especificar al sistema órdenes que no son propias del programa, pero son necesarias para correrlo.

## K

### **Kernel:**

«núcleo»; La parte central de un sistema operativo, sobre la cual el resto del sistema se apoya.

## L

### **Llamada al sistema:**

Es una rutina que cumple una función a nivel del sistema en nombre de un proceso. En inglés: «System Call».

### **Linux con seguridad mejorada:**

Abreviatura para Security Enhanced Linux o Seguridad mejorada de Linux, SELinux utiliza módulos de seguridad mejorada de Linux (LSM) en el kernel de Linux para proporcionar un rango de políticas de seguridad de mínimo privilegio requerido.

### **LUN:**

Un número de unidad lógica (LUN) del inglés Logical Unit Number es un número asignado a una unidad lógica (una entidad de protocolo SCSI).

## M

### **Máquina Virtual:**

Se trata de un tipo de sistemas operativos que presentan una interface a cada proceso, mostrando una máquina que parece idéntica a la máquina real subyacente.

### **Maquinas virtual por hardware:**

Consulte [Virtualización completa](#)

### **Máquinas virtuales:**

Una máquina virtual es una implementación software de una máquina física o un lenguaje de programación (por ejemplo, el entorno en tiempo de ejecución Java o LISP). Las máquinas virtuales en el contexto de virtualización son sistemas operativos en hardware virtualizado.

### **Máquinas virtuales basadas en el Kernel:**

KVM (Kernel-based Virtual Machine, Máquina Virtual basada en el Kernel) es una Virtualización completa solución para Linux sobre hardware AMD64 e Intel 64. VM es un módulo del Kernel de Linux construido para el Kernel estándar de Linux para empresas de Red Hat. KVM puede ejecutar diferentes sistemas operativos huéspedes, ya sean Windows o Linux, sin tener que modificarlos. KVM es un hipervisor que utiliza las herramientas de virtualización libvirt (virt-manager y virsh).

KVM es un conjunto de módulos de kernel de Linux que administran dispositivos, memoria y APIs de administración para el módulo Hipervisor en sí mismo. Los invitados virtualizados corren como procesos Linux y threads que se controlan con estos módulos.

### **Máscara:**

Un Mecanismo creado para el control de las interrupciones.

### **Mecanismo de estados:**

Mecanismo ideado para conservar la consistencia e integridad en un sistema multiusuario.

### **Memoria virtual:**

La idea fundamental detrás de la memoria virtual es que el tamaño combinado del programa, los datos y la pila de ejecución puede exceder la cantidad de memoria real disponible para él. El sistema operativo mantiene aquellas partes del programa que están utilizando en cada momento en la memoria principal y el resto permanece en el disco. En la medida en que se vayan necesitando nuevas partes estas se intercambian con las residentes en la memoria principal.

### **Memoria:**

Cuando se refiere a ordenadores, la memoria (en general) es cualquier hardware capaz de almacenar datos para recuperarlos posteriormente. En este contexto, la memoria en general se refiere específicamente a la RAM.

### **Migración:**

Migración es el nombre que recibe el proceso de desplazar un huésped virtualizado de un equipo hacia otro. La migración puede realizarse desconectado (cuando el huésped se interrumpe y luego se traslada), o conectado o en vivo (cuando el huésped es desplazado y trasladado sin interrupción). Los huéspedes KVM totalmente virtualizados pueden ser migrados.

La migración es una función clave de la virtualización ya que el software es totalmente independiente del hardware. La migración es útil para:

- Equilibrio de carga - los huéspedes pueden ser trasladados hacia equipos de menor utilización cuando alguno se encuentre sobrecargado.
- Conmutación por error de hardware - cuando algún dispositivo de hardware del equipo empiece a funcionar erróneamente, los huéspedes pueden ser realojados en forma segura, de modo de permitir que el equipo

problemático sea desconectado y reparado.

- Ahorro de energía - los huéspedes pueden ser redistribuidos hacia otros equipos, de modo de permitir desconectar los equipos originarios para ahorrar energía, y abaratar costos en períodos de escasa utilización.
- Migración geográfica - los huéspedes pueden ser trasladados hacia otras ubicaciones para disminuir la latencia, o por circunstancias de fuerza mayor.

Almacenamiento compartido o en redes se utiliza para almacenar imágenes de huésped.

Sin migración de almacenamiento compartido esto no es posible.

#### **Monitores de encadenamiento:**

Constituyeron los primeros sistemas operacionales. Con éstos se introdujo también el llamado tratamiento por lotes (batch processing).

#### **Multiprogramación:**

Se caracteriza por una multitud de programas activos simultáneamente que compiten por los recursos del sistema, tales como procesador, memoria y dispositivos de E/S.

#### **Multitarea:**

Un solo usuario ejecutando varias tareas al mismo tiempo en un sistema.

#### **N**

#### **Núcleo:**

La parte central de un sistema operativo, sobre la cual el resto del sistema se apoya. En inglés: «kernel».

#### **P**

#### **Paso a través de PCI:**

El hipervisor KVM ofrece soporte para asociar dispositivos PCI en el sistema del equipo anfitrión para poder virtualizar huéspedes. El paso a través de PCI permite que los huéspedes tengan acceso exclusivo a dispositivos PCI para una gran variedad de tareas. El paso PCI permite que los

dispositivos PCI aparezcan y se comporten como si se encontraran físicamente asociados con el sistema operativo huésped.

#### **PCB:**

(Process Control Bloc). Estructura de datos almacenan la PEP, aspectos sobre la ejecución del proceso y de la utilización de los recursos.

#### **Planificación:**

Proceso cuyo objetivo principal es repartir el tiempo del procesador entre los procesos ejecutables y optimizar el comportamiento del sistema.

#### **Prioridad:**

Número asociado a cada proceso o thread y utilizado por la política de planificación.

#### **Procesador:**

Llamado también Microprocesador. Es el chip encargado de ejecutar las instrucciones y procesar los datos que son necesarios para todas las funciones de la computadora. Se puede decir que es el cerebro de la computadora. El estándar del mercado es el fabricado por la empresa INTEL.

#### **Proceso:**

programa en ejecución.

#### **Programa:**

Es esencialmente un archivo ejecutable que se coloca en memoria para que pueda el procesador ir tomando su código, y obedeciendo las instrucciones de él.

#### **PSW-PEP:**

(Program status Word o Palabra de Estado del Programa); Contiene toda la información del estado del proceso y define sus privilegios.

#### **S**

#### **SGMV:**

Sistema Generador de Máquina Virtual.

#### **Shell:**

Intérprete de comandos.

#### **Sincronización:**

Necesidad que tiene algunos procesos de bloquearse en determinadas circunstancias y ser despertados cuando ocurren ciertos eventos.

**Sistema de Archivos o Sistema de ficheros:**

Es el método mediante el cual se almacena la información en las unidades de disco. Los distintos sistemas operativos normalmente usan diferentes sistemas de ficheros, lo que dificulta el compartir los contenidos de una unidad de disco entre ellos. Sin embargo, Linux admite múltiples sistemas de ficheros, lo cual hace posible la lectura/escritura de particiones dedicadas a MS-Windows, por ejemplo. En inglés: «Filesystem».

**Sistema Monolítico:**

Sistemas que tienen un núcleo grande y complejo, que engloba todos los servicios del sistema.

**Sistema Monousuario:**

Sistemas operativos que no pueden dar servicio sino sólo a un usuario cada vez, sin importar el número de procesadores que tenga la computadora o el número de tareas o procesos que el usuario pueda ejecutar en un instante de tiempo.

**Sistema Multiusuario:**

Sistemas que pueden atender múltiples usuarios, ya sea por medio de varias terminales conectadas a la computadora o por medio sesiones remotas en una red de comunicaciones.

**Sistema huésped:**

También conocidas como huéspedes, máquinas virtuales, servidores virtuales o dominios.

**Sistema operativo:**

Conjunto de software que controla los distintos recursos del ordenador. En inglés: «operating system».

**Sistemas de tiempo compartido:**

Son sistemas de multiprogramación en donde existe interacción directa entre el usuario y la máquina.

**Spool:**

Virtualización de lectoras e impresoras.



**Virtualización**

Virtualización es un término general de informática aplicado a la ejecución de software (generalmente sistemas operativos), al mismo tiempo que se ejecutan otros programas en el sistema, y de manera aislada de ellos. La mayoría de las implementaciones de virtualización actuales, utilizan un hipervisor, esto es, una capa de software colocada sobre un sistema operativo de modo de poder abstraer un determinado hardware. El hipervisor permite que múltiples sistemas operativos sean ejecutados en el mismo sistema físico, al brindarle hardware virtualizado al sistema operativo huésped. Existen varios métodos para virtualizar sistemas operativos:

- Una virtualización asistida por hardware es la técnica utilizada para virtualizaciones completas mediante KVM (definición: Virtualización completa)

**Virtualización completa:**

KVM utiliza una virtualization completa y asistida por hardware. La virtualización completa utiliza las funciones del hardware del procesador para proporcionar abstracción total del sistema físico subyacente (Bare-metal), y crea una nueva máquina virtual en el cual los sistemas operativos huéspedes puedan ejecutarse. No se necesita ninguna modificación en el sistema operativo

huésped. El sistema operativo huésped y cualquiera de sus aplicaciones no son conscientes del entorno virtualizado, y se ejecutan normalmente. Una para-

virtualization requiere una versión modificada del sistema operativo Linux.



## Referencias bibliográficas

ANGELES, S. 2014. Virtualization vs. Cloud Computing: What's the Difference? Disponible en: <https://www.businessnewsdaily.com/5791-virtualization-vs-cloud-computing.html>

IBM.Soluciones para la implantación de la virtualización de servidores –IBM Emerging Server Technology Services. San Fernando de Henares, Madrid: s.n., 2007. Disponible en: [http://www05.ibm.com/services/es/ss/pdf/Server\\_Virtualization.pdf](http://www05.ibm.com/services/es/ss/pdf/Server_Virtualization.pdf)

CISCO Systems Inc.El ROI de la virtualización: Obtenga fabulosos resultados con sus servidores. 2011. Disponible en: [http://www.cisco.com/web/LA/soluciones/comercial/centro\\_recursos/pdf/El\\_ROI\\_de\\_la\\_virtualizacion.pdf](http://www.cisco.com/web/LA/soluciones/comercial/centro_recursos/pdf/El_ROI_de_la_virtualizacion.pdf)

CISCO Systems Inc.Cuatro pasos para la virtualización: Cómo comenzar un proyecto de servidores. 2011. Disponible en: [http://www.cisco.com/web/LA/soluciones/comercial/centro\\_recursos/pdf/Cuatro\\_pasos\\_para\\_la\\_virtualizacion.pdf](http://www.cisco.com/web/LA/soluciones/comercial/centro_recursos/pdf/Cuatro_pasos_para_la_virtualizacion.pdf)

Villar, Eugenio y Gómez, Julio. Virtualización de servidores de telefonía ip en gnu/Linux.Trabajo de grado de Ingeniero en Informática.Almería, España: Universidad de Almería,2010. 324 p. Disponible en [http://www.adminso.es/recursos/Proyectos/PFC/PFC\\_eugenio.pdf](http://www.adminso.es/recursos/Proyectos/PFC/PFC_eugenio.pdf) [Citado en 25 de julio de 2012]

F5 Networks, Data Center Virtualization Q&A. Disponible en: <http://www.f5.com/pdf/white-papers/dc-virtualization-wp.pdf> [Citado en 19 de septiembre de 2012]