



UNIVERSIDAD DEL AZUAY

**FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN
ESCUELA DE INGENIERÍA DE SISTEMAS**

TEMA:

**“Configuración, Creación Y Supervisión de Sistemas Operativos Huéspedes,
mediante el Hypervisor Xen”.**

Monografía previa a la obtención del título de:
Ingeniero en Sistemas.

Autor:

Juan Pablo Huiracocha Piedra.

Director:

Ing. Pablo Esquivel.

Tribunal:

Ing. Katherine Ortiz.

CUENCA ECUADOR
2011

DEDICATORIA

Este trabajo de graduación lo dedico a mi familia que gracias a su apoyo pude culminar con éxito mi carrera. A mis padres y hermanos por su apoyo y confianza. Gracias por ayudarme alcanzar mis objetivos como persona y estudiante. A mi padre por brindarme los recursos necesarios y estar a mi lado apoyándome y aconsejándome siempre. A mi madre por hacer de mí una mejor persona a través de sus consejos, enseñanzas y amor. A mis hermanos por estar siempre presentes, acompañándome para poderme realizar.

AGRADECIMIENTO

A mis profesores quienes me han enseñado a ser mejor en la vida y a realizarme profesionalmente, un agradecimiento especial a la Universidad del Azuay por brindarme la oportunidad de realizar los estudios de Ingeniería de sistemas.

Índice de Contenidos

Dedicatoria.....	ii
Agradecimientos.....	iii
Índice de Contenidos.....	iv
Resumen.....	vii
Abstract.....	viii

CAPITULO 1

VIRTUALIZACIÓN

1.1 Introducción.....	1
1.2 Concepto de virtualización.....	1
1.3 Tipos de virtualización.....	2
1.3.1 Emulación.....	3
1.3.2 Virtualización Completa.....	3
1.3.3 Paravirtualización.....	5
1.3.4 Virtualización a nivel de sistema operativo.....	6
1.4 Máquinas virtuales.....	7
1.5 EL Virtual Machine Monitor (VMM).....	16
1.6 Protocolos de comunicación de equipos virtualizados.....	17
1.7 Clasificando la Virtualización.....	18
1.7.1 Virtualización de hardware.....	19
1.7.2 Virtualización de sistemas operativos.....	19
1.7.3 Virtualización de Aplicaciones.....	20
1.7.4 Virtualización de equipos de redes.....	20
1.8 Problemas y posibles soluciones en la Virtualización.....	21

CAPITULO 2

ARQUITECTURA DEL SISTEMA DE VIRTUALIZACIÓN XEN

2.1 Introducción.....	22
2.2 Que es Xen?.....	22
2.3 Que es el Hypervisor.....	25
2.4 Sistemas operativos soportados.....	26
2.5 Hardware soportado.....	27
2.6 El dom0 y los domU.....	28
2.7 Requisitos de Instalación.....	28

CAPITULO 3

XEN

3.1	Introducción.....	30
3.2	Instalación Xen.....	30
3.3	Configuración GRUB.....	41
3.4	Discos Virtuales.....	44
3.5	La herramienta kpartx.....	46
3.6	Configuración LVM y Raid.....	61
3.7	Administración de máquinas virtuales con xm.....	66
3.7.1	Instalación de un huésped centos 5.5 con el método paravirtual.....	67
3.7.2	Instalación de un huésped windows con método fullvirtual.....	81
3.8	Logs en xen.....	92
3.9	Migración de máquinas virtuales.....	93
3.10	Inicio automático de los domU.....	99
3.11	Tarjetas de red en XEN.....	100
3.12	Network XEN.....	103
3.13	Herramienta brctl.....	108

CAPITULO 4

ADMINISTRACIÓN DE MÁQUINAS VIRTUALES CON VIRSH

4.1	Introducción.....	111
4.2	Conexión al hypervisor.....	113
4.3	Creación de una máquina virtual.....	113
4.4	Configuración de un volcado xml.....	114
4.5	Suspensión de una máquina virtual.....	116
4.6	Reanudación de una máquina virtual.....	116
4.7	Guardar una máquina virtual.....	117
4.8	Apagado de una máquina virtual.....	117
4.9	Convertir un nombre de dominio a un ID de dominio.....	117
4.10	Ver información de una máquina virtual.....	117

CAPITULO 5

ADMINISTRACIÓN DE MÁQUINAS VIRTUALES CON VIRT-MANAGER

5.1	Introducción.....	118
5.2	Ventana para administrador Máquinas Virtuale.....	118
5.3	Ventana de detalles de la Máquina Virtual.....	119

5.4	Consola gráfica de la Máquina Virtual	120
5.5	Crear una nueva Máquina Virtual.....	121

CAPITULOS 6

DETECCIÓN Y CORRECCIÓN DE ERRORES

6.1	Introducción.....	129
6.2	Descripción de los archivos de registro.....	130
6.3	Detección y resolución de errores.....	131
6.4	Herramientas para la detección y solución de problemas.....	132
6.5	Detección y solución de errores con la consola serial.....	134

CAPITULO 7

CONCLUSIONES Y RECOMENDACIONES

7.1	Conclusiones.....	135
7.2	Recomendaciones.....	135

GLOSARIO.....	138
----------------------	------------

BIBLIOGRAFÍA.....	140
--------------------------	------------

RESUMEN

Es la Virtualización una forma de administrar los equipos y aplicaciones de una organización, permitiendo una fácil administración y control de estos, es una forma de diseñar y administrar la red.

Una herramienta para realizar este trabajo es el Hypervisor Xen, permite entre otras cosas la virtualización de escritorios y aplicaciones, aspectos que en este trabajo son abordados desde la arista de la implementación de estos.

En la actualidad el tema de administración de recursos y una armonía con el medio ambiente van de la mano con la tecnología es por eso que es, la virtualización una herramienta a tener en cuenta al momento de diseñar un entorno informático.

En el presente trabajo se implementa un laboratorio de pruebas y se realizan mediciones de acceso y de consumo que permiten entre otras cosas obtener una visión del trabajo, la implementación, los costos asociados y las significancias al usuario al utilizar la virtualización como medio de trabajo.

ABSTRACT

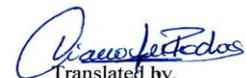
Virtualization is a way to administer the equipments and applications of an institution. It allows managing and controlling them and it is a way of designing and managing the web.

Xen Hypervisor is a tool used for this type of work; it permits the application and desktop virtualization, which are the aspects analyzed in this project.

Nowadays, management of resources and environmental care go hand in hand with technology and for this reason, virtualization is a tool that has to be considered at the moment of designing an informatics system.

The present research project is implemented in a testing laboratory and measurements of the entrées and consumptions are carried out, which allow us to visualize the work, the implementation, the associated costs and user significance while employing virtualization as a working tool.




Translated by,
Diana Lee Rodas

The text block contains a handwritten signature in blue ink, followed by the text "Translated by," and the name "Diana Lee Rodas" in a standard font.

CAPITULO 1

VIRTUALIZACIÓN

1.1 Introducción

La tecnología nos ofrece la posibilidad de disponer de varios servidores con características muy distintas pero instalando físicamente una sola máquina. Este sistema se llama virtualización. A través de esta tecnología es posible hacer que los recursos de un ordenador, en este caso un servidor, puedan ser compartidos por una o más máquinas virtuales que se comportarán a su vez como servidores reales.

La virtualización está de moda. Si antes era una tecnología al alcance solamente de las grandes compañías. Ahora cualquier empresa pequeña o mediana tiene al alcance los beneficios de la virtualización.

1.2 Concepto de virtualización.

“La virtualización hace referencia a la ocultación o abstracción de recursos de una computadora”. [VWK2011]. La abstracción de recursos se lo conoce como VMM (Virtual Machine Monitor) o monitor de máquina virtual que crea una capa de abstracción entre el hardware de la máquina física o host y el sistema operativo de la máquina virtual o guest, siendo un medio para crear una versión virtual de un dispositivo o recurso, como dispositivos de red, discos de almacenamiento, en donde el recurso se dividió, en uno o más recursos.

El Virtual Machine Monitor se encarga de gestionar los principales recursos de una computadora estos son cpu, memoria, almacenamiento, red y así repartir dinámicamente los recursos aplicando técnicas de balanceo de carga,

permitiendo ejecutar varios sistemas operativos virtuales, ejecutándose sobre el mismo ordenador físico.

Lo más interesante del término virtualización, es la de ocultar detalles técnicos a través de la encapsulación.

Hablamos de virtualización, cuando es 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 hypervisor, esto es, una capa de software, colocada sobre un sistema operativo, de modo que se abstraen un determinado hardware, esto 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:

- Virtualización completa.
- Para-virtualización.
- Virtualización de software o emulación.

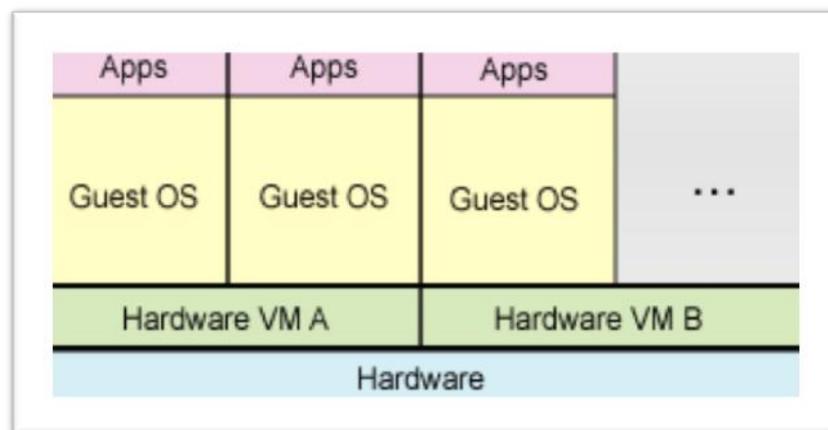
1.3 Tipos de virtualización.

La virtualización se lleva a cabo en una plataforma de hardware mediante un software "host", que es un programa de control que simula un entorno computacional (máquina virtual) para su software "guest" (en castellano "huésped"). [VWP2011]. Este software "huésped", que generalmente es un sistema operativo completo, se ejecuta como si estuviera instalado en una plataforma de hardware autónoma. Típicamente muchas máquinas virtuales son simuladas en una máquina física dada. Para que el sistema operativo "huésped" funcione, la simulación debe ser lo suficientemente grande como para soportar todas las interfaces externas de los sistemas huéspedes, las

cuales pueden incluir (dependiendo del tipo de virtualización) los drivers de hardware.

1.3.1 Emulación.

La emulación se basa en crear máquinas virtuales que emulan el hardware de una o varias plataformas hardware distintas. [XUM2008]. Este tipo de virtualización es la más costosa y la menos eficiente, ya que obliga a simular completamente el comportamiento de la plataforma hardware e implica también que cada instrucción que se ejecute en estas plataformas sea traducida al hardware real y es demasiado lento en su ejecución. En Linux existen varios emuladores pero los más conocidos son: Qemu, Bochs, VirtualBox y Virtuozzo.



1.3.2 Virtualización Completa.

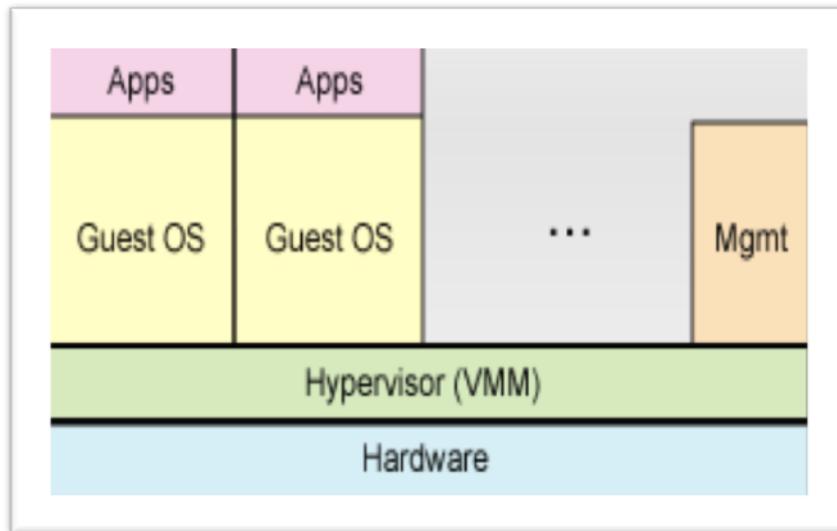
Con este término se denominan aquellas soluciones que permiten ejecutar sistemas operativos huésped (Guest), sin tener que modificarlos, sobre un sistema anfitrión (Host), utilizando en medio un Hypervisor o Virtual Machine Monitor que permite compartir el hardware real. [XUM2008]. Esta capa intermedia es la encargada de monitorizar los sistemas huésped con el fin de capturar determinadas instrucciones protegidas de acceso al hardware, que no pueden realizar de forma nativa al no tener acceso directo a él.

Su principal ventaja es que los sistemas operativos pueden ejecutarse sin ninguna modificación sobre la plataforma, aunque como inconveniente frente a la emulación, el sistema operativo debe estar soportado en la arquitectura virtualizada.

En lo que respecta al rendimiento, éste es significativamente mayor que en la emulación, pero menor que en una plataforma nativa, debido a la monitorización y la mediación del hypervisor. Sin embargo, recientes incorporaciones técnicas en las plataformas x86 hechas por Intel y AMD, como son Intel VT y AMD-V, han permitido que soluciones basadas en la virtualización completa se acerquen prácticamente al rendimiento nativo.

Hay que tener en cuenta también que la virtualización completa no se refiere a todo el conjunto de hardware disponible en un equipo, sino a sus componentes principales, básicamente el procesador y memoria. De esta forma, otros periféricos como tarjetas gráficas, de red o de sonido, no se virtualizan. Las máquinas huésped no disponen de los mismos dispositivos que el anfitrión, sino de otros virtuales genéricos. Por ejemplo, si se dispone de una tarjeta NVidia GeForce en el anfitrión, los equipos huésped no verán esta tarjeta sino una genérica Cirrus.

Existen herramientas para poder hacer una virtualización completa entre ellos constan xen, kvm, local domains y vmware.



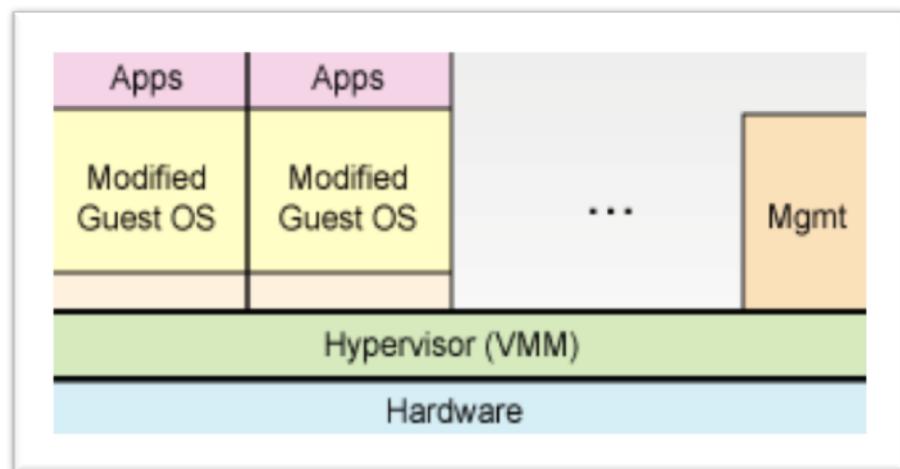
1.3.3 Paravirtualización.

Paravirtualización surgió como una forma de mejorar la eficiencia de las máquinas virtuales y acercarlo al rendimiento nativo. [XUM2008]. Para ello se basa en que los sistemas virtualizados huésped deben estar basados en sistemas operativos especialmente modificados para ejecutarse sobre un hypervisor, esto quiere decir que el huésped debe permitir modificar su kernel para adaptarlo al hypervisor. Todo este trabajo se lleva a cabo en la carga del procesador, ya que tiene que tener incluido el soporte de virtualización. Se pueden ocupar las herramientas de xen, kvm y vmware.

La finalidad de modificar el kernel de los huéspedes es reducir la porción del tiempo de ejecución del usuario, empleado en operaciones que son sustancialmente más difíciles de ejecutar, en un entorno virtual, comparado con un entorno no virtualizado. La paravirtualización, provee filtros especialmente definidos para permitir a los invitados y al anfitrión hacer peticiones y conocer estas tareas, que de otro modo serían ejecutadas en el dominio virtual (donde el rendimiento de la ejecución es peor). Por lo tanto, una plataforma de paravirtualización exitosa puede permitir que el monitor de la máquina virtual (VMM) sea más simple, por traslado de la ejecución de tareas críticas desde el dominio virtual al anfitrión de dominio, y/o que

reduzca la degradación del rendimiento global de la ejecución de la máquina dentro del anfitrión virtual.

La paravirtualización requiere que el sistema operativo invitado sea portado de manera explícita para la API. Una distribución de un sistema operativo convencional que no soporte paravirtualización, no puede ser ejecutada ni visualizada en un monitor de máquina virtual VMM

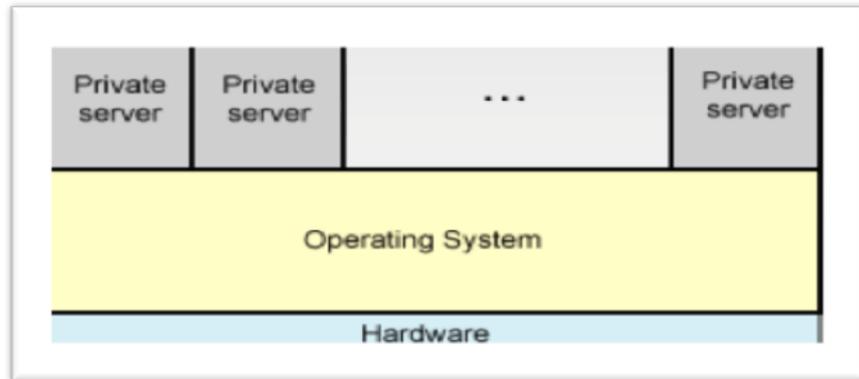


1.3.4 Virtualización a nivel de sistema operativo

El concepto de una virtualización a nivel de sistema operativo se basa en compartir, el mismo sistema operativo que el hospedero. En verdad todos corren el mismo kernel y es el kernel el que se ocupa de determinar para quién trabaja en un momento determinado. Los huéspedes se ven como si estuvieran solos pero en verdad todos comparten el mismo kernel, cosa que hace que una vulnerabilidad en el kernel afecte a todos. [XUM2008].

La virtualización de SO mejora el rendimiento, gestión y eficiencia. En la base reside un sistema operativo huésped estándar, como en el caso de Parallels Virtuozzo, que incluye Windows y Linux. A continuación encontramos la capa de virtualización, con un sistema de archivos propietario y una capa de abstracción de servicio de kernel que garantiza el aislamiento y seguridad de los recursos entre distintos contenedores. La capa de

virtualización hace que cada uno de los contenedores aparezca como servidor autónomo. Finalmente, el contenedor aloja la aplicación o carga de trabajo.



1.4 Máquinas virtuales

Las máquinas virtuales son una de las opciones de virtualización existentes. Una definición que se acerca bastante es la de Goldberg (1974), que la define como “un duplicado de hardware y software del sistema de computación real en el cual un subconjunto de instrucciones del procesador se ejecuta sobre el procesador anfitrión en modo nativo”.

En informática una máquina virtual es un software que emula a una computadora y puede ejecutar programas como si fuese una computadora real. Este software en un principio fue definido como "un duplicado eficiente y aislado de una máquina física". [VWK2011]. La acepción del término actualmente incluye a máquinas virtuales que no tienen ninguna equivalencia directa con ningún hardware real.

Una característica esencial de las máquinas virtuales es que los procesos que ejecutan están limitados por los recursos y abstracciones proporcionados por ellas. Estos procesos no pueden escaparse de esta "computadora virtual".

Uno de los usos domésticos más extendidos de las máquinas virtuales es ejecutar sistemas operativos para "probarlos". De esta forma podemos ejecutar un sistema operativo que desea probar por ejemplo GNU/Linux,

desde un sistema operativo habitual (Mac OS X por ejemplo) sin necesidad de instalarlo directamente en nuestra computadora y sin miedo a que se desconfigure el sistema operativo primario.

A las máquinas virtuales las podemos clasificar de la siguiente manera.

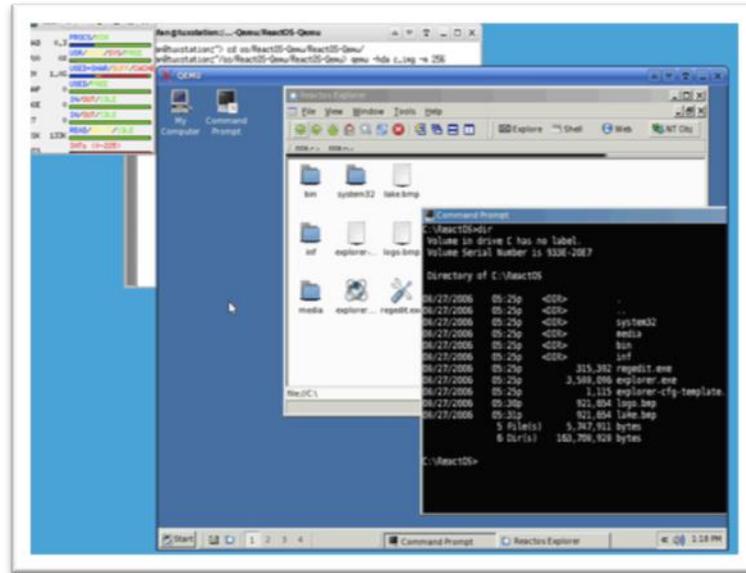
Emulación:

Qemu.

QEMU es un emulador de procesadores basado en la traducción dinámica de binarios; conversión, del código binario de la arquitectura fuente, en código entendible por la arquitectura huésped. [QMU2011]. QEMU también tiene capacidades de virtualización dentro de un sistema operativo, ya sea GNU/Linux, Windows, o cualquiera de los sistemas operativos admitidos. Esta máquina virtual puede ejecutarse en cualquier tipo de Microprocesador o arquitectura (x86, x86-64, PowerPC, MIPS, SPARC, etc.). Está licenciado en parte con la LGPL y la GPL de GNU.

El objetivo principal, es emular un sistema operativo dentro de otro, sin tener que reparticionar el disco duro; empleando para su ubicación cualquier directorio dentro de éste.

El programa no dispone de GUI; pero existe otro programa llamado QEMU manager, que hace las veces de interfaz gráfica, si se utiliza QEMU desde Windows. También existe una versión para GNU/Linux llamado qemu-launcher. En Mac OS X, puede utilizarse el programa Q, que dispone de una interfaz gráfica para crear y administrar las máquinas virtuales.



Bochs.

Bochs (pronunciado "box") es un emulador de código abierto para PC, escrito en C++ por Kevin Lawton. Es capaz de ejecutarse en múltiples plataformas y diferentes arquitecturas. Se distribuye bajo la licencia GPL. [BCH2010].

Bochs posee la capacidad de emular una CPU con arquitectura Intel x86, dispositivos comunes de E/S, y un BIOS. Actualmente, Bochs puede ser compilado para emular un i386, i486 Pentium/Pentium II/Pentium III/Pentium 4 o una CPU con arquitectura x86-64, incluyendo instrucciones adicionales como las MMX, SSEx y 3Dnow. [BCH2010].

Además, es capaz de ejecutar una gran variedad de sistemas operativos, dentro de su emulación, entre los cuales están: Linux, DOS, Windows 95/98, Windows NT/2000/XP y Windows Vista. Bochs, es muy utilizado en la depuración de sistemas operativos, ya que cuando el sistema operativo huésped cae por alguna razón, el sistema operativo anfitrión no cae también, lleva un registro de errores y de volcado de archivos, adicional a esto, Bochs emula todo el hardware que ocupa el sistema operativo huésped, emulando discos duros, cdrom, disquetes, es posible insertar, mientras está corriendo el

emulador, estas unidades, esto ocasiona un efecto negativo en el rendimiento, incluso siendo más lento que el emulador QEMU. [BCH2010].

A diferencia de otros emuladores más populares, su interfaz gráfica de usuario es menos intuitiva, pero ofrece opciones de personalización editando el fichero bochssrc.txt.



VirtualBox.

Oracle VM VirtualBox, es un software de virtualización para arquitecturas x86, creado originalmente por la empresa alemana innotek GmbH. Actualmente es desarrollado por Oracle Corporation, como parte de su familia de productos de virtualización. 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. [VTB2011].

Entre los sistemas operativos soportados en modo anfitrión, se encuentran GNU/Linux, Mac OS X, OS/2 Warp, Microsoft Windows, y Solaris/OpenSolaris, y dentro de ellos es posible virtualizar los sistemas

operativos FreeBSD, GNU/Linux, OpenBSD, OS/2 Warp, Windows, Solaris, MS-DOS y muchos otros. [VTB2011].

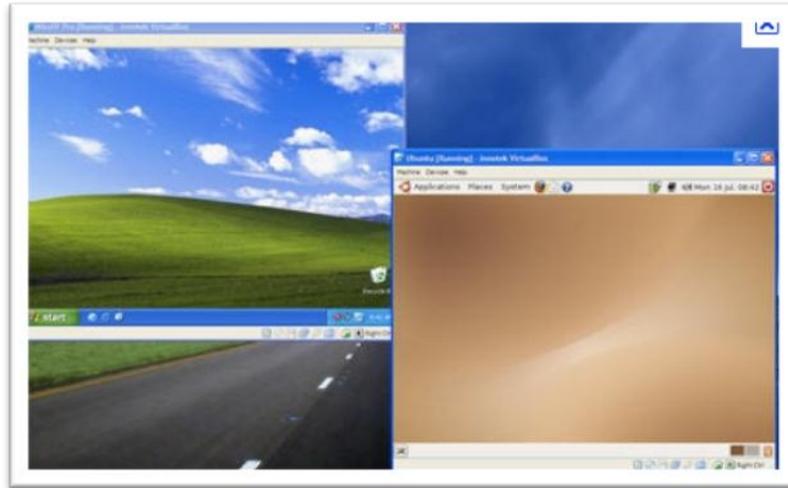
La aplicación fue inicialmente ofrecida bajo una licencia de software propietario, pero en enero de 2007, después de años de desarrollo, surgió VirtualBox OSE (Open Source Edition) bajo la licencia GPL 2. Actualmente existe la versión propietaria Oracle VM VirtualBox, que es gratuita únicamente bajo uso personal o de evaluación, y está sujeta a la licencia de "Uso Personal y de Evaluación VirtualBox" (VirtualBox Personal Use and Evaluation License o PUEL) y la versión Open Source, VirtualBox OSE, que es software libre, sujeta a la licencia GPL. [VTB2011].

VirtualBox, ofrece algunas funcionalidades interesantes, como la ejecución de máquinas virtuales de forma remota, por medio del Remote Desktop Protocol (RDP), soporte iSCSI, aunque estas opciones no están disponibles en la versión OSE. [VTB2011].

En cuanto a la emulación de hardware, los discos duros de los sistemas invitados, son almacenados en los sistemas anfitriones, como archivos individuales en un contenedor llamado Virtual Disk Image, incompatible con los demás software de virtualización. [VTB2011].

Otra de las funciones que presenta es la de montar imágenes ISO como unidades virtuales ópticas de CD o DVD, o como un disquete.

Tiene un paquete de controladores que permiten aceleración en 3D, pantalla completa, hasta 4 placas de red Gigabit, integración con teclado y mouse. [VTB2011].

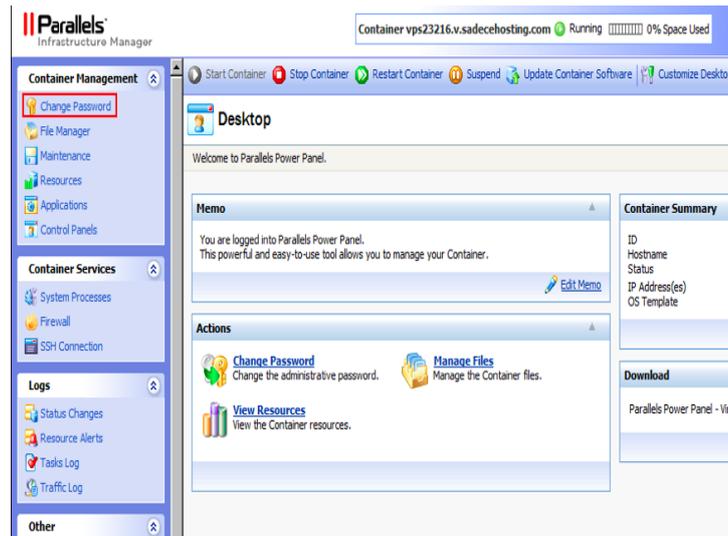


Virtuozzo.

Virtuozzo crea múltiples entornos virtuales independientes (VEs, también conocidos como servidores privados virtuales, o VPSs) en un solo servidor físico. Esto permite compartir hardware, tiempo y esfuerzo en la administración de sistemas y a veces, hasta las licencias del software entre los múltiples entornos virtuales. [VTZ2010].

Los VPS actúan en la mayoría de los casos como si fueran un servidor individual. Cada VPS tiene su propio súper usuario (raíz o administrador), sistema de usuarios/de grupos, dirección(es) ip, procesos, archivos, usos, bibliotecas de sistema y archivos de configuración. Los VPS son totalmente accesibles vía red. Debido a la creación de diferentes servidores de red virtualizados, un propietario de VPS puede tener sus propias tablas de enrutamiento y reglas de seguridad en el firewall (IPtables). [VTZ2010].

Virtuozzo, puede crear, decenas o centenares de VPS, en un solo servidor debido a su forma de funcionamiento y realizar la virtualización a nivel sistema. Está disponible para Linux y Windows. [VTZ2010].

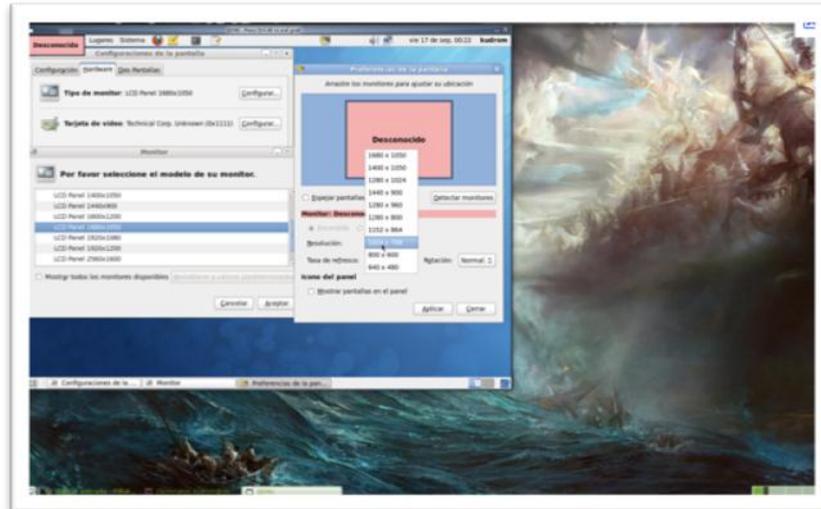


Virtualización completa:

Xen

Xen, es un monitor de máquina virtual de código abierto, desarrollado por la Universidad de Cambridge. [XWP2011].

La meta del diseño es poder ejecutar instancias de sistemas operativos con todas sus características, de forma completamente funcional en un equipo sencillo. Xen proporciona aislamiento seguro, control de recursos, garantías de calidad de servicio y migración de máquinas virtuales en caliente. Los sistemas operativos pueden ser modificados explícitamente para correr Xen (aunque manteniendo la compatibilidad con aplicaciones de usuario). Esto permite a Xen alcanzar virtualización de alto rendimiento sin un soporte especial de hardware. Intel ha realizado diversas contribuciones a Xen que han permitido añadir soporte para sus extensiones de arquitectura VT-X Vanderpool. Esta tecnología permite que sistemas operativos sin modificar actúen como hosts dentro de las máquinas virtuales de Xen, siempre y cuando el servidor físico soporte las extensiones VT de Intel o Pacifica de AMD. [XWP2011].



Vmware esx.

VMware ESX, es un hypervisor del tipo "bare metal" En oposición a otros productos, no se ejecuta sobre un sistema operativo externo sino que está embebido en el núcleo del mismo. Para su ejecución, ESX se apoya en un sistema Linux, basado en Red Hat Enterprise Linux modificado para la ejecución del hypervisor y los componentes de virtualización de VMware. Hasta la versión 3.5u4 se basa en código ejecutable de 32 bits, pero a partir de la versión 4 (vSphere) su código ejecutable pasa a ser de 64 bits, por lo que sus requerimientos pasan a ser mayores ofreciendo a su vez un rendimiento superior. [VMW2011].

El arranque de la máquina anfitriona se produce a través de la ejecución de un núcleo Linux, el cual proporciona servicios de consola y hardware. [VMW2011].

A partir de la versión vSphere (versión 4.0), el hypervisor aplica los teoremas de la paravirtualización y sustituye al núcleo Linux por sus propias interfaces, y pasando a ejecutar el entorno operativo como una máquina virtual. [VMW2011].

El vmkernel está programado y configurado siguiendo la arquitectura de microkernel, y tiene las siguientes interfaces con el exterior:

- hardware
- sistema invitado
- consola de servicio (Console OS, Service Console)



1.5 EL Virtual Machine Monitor (VMM)

Hypervisor o Virtual Machine Monitor (VMM), es una tecnología que está compuesta por una capa de software que permite utilizar al mismo tiempo, diferentes sistemas operativos o máquinas virtuales (sin modificar o modificados en el caso de paravirtualización) en una misma computadora central; es decir es la parte principal de una máquina virtual, se encarga de manejar los recursos del sistema principal, exportándolos a la máquina virtual. [HYP2007].

El VMM (Virtual Machine Monitor), crea una capa de la abstracción, entre el hardware de la máquina física (host) y el sistema operativo de la máquina virtual (virtual machine, guest), de tal forma, que maneja los recursos de las máquinas físicas subyacentes (designadas por el computador central) de una manera que el usuario pueda crear varias máquinas virtuales, presentando a cada una de ellas una interfaz del hardware que sea compatible con el sistema operativo elegido. [HYP2007].

Esta capa de software (VMM), maneja, gestiona y arbitra los cuatro recursos principales de una computadora: cpu, memoria, red, almacenamiento y así, podrá repartir dinámicamente dichos recursos entre todas las máquinas virtuales definidas en el computador central. [HYP2007].

En la actualidad, todos los fabricantes tanto de Software como de Hardware están trabajando para mejorar y ayudar al Hypervisor (VMM) y así poder llegar a una virtualización completa, fiable y robusta. [HYP2007].

1.6 Protocolos de comunicación de equipos virtualizados

Existen diferentes tipos de protocolos para permitir el acceso a la máquina virtual, a través de un cliente ligero o interfaz de comunicación remota cualquiera sea la que se ocupe.

La manera de comunicarse más común y que hace uso de un protocolo llamado vnc que es un protocolo open source y que permite un acceso por medio de un modo cliente servidor de vnc.

Otro protocolo de comunicación es la tecnología nx, creada por la empresa italiana no machine, que permite la comunicación encriptada a través de túneles ssh; vmware tiene su protocolo propio para realizar la conexión remota de equipos, PCoIP (pc over ip) el cual permite la comunicación entre equipos remoto y se destaca por transferir exclusivamente en UDP para proporcionar mapas de bits.

En Citrix Xen que será la plataforma de pruebas, trabaja con el protocolo ICA para realizar el trabajo de enviar la imagen del escritorio virtual hacia el thinclient, en donde solo se envía a través de tcp a diferencia de PcoIP; permite la utilización de una tecnología llamada flexcast, la cual permite la utilización de cualquier método de virtualización de escritorio en cualquier dispositivo; Además esta tecnología permite ejecutar de manera offline las aplicaciones.



1.7 Clasificando la Virtualización.

La virtualización se puede clasificar según distintos tipos de vista, se clasificará para ser más específicos y para que se tenga en claro bajo que ámbitos está presente la virtualización, entonces bajo este criterio se clasifica en:

- Virtualización de hardware.
- Virtualización de sistema operativo.
- Virtualización de aplicaciones.
- Virtualización de equipos de red.

1.7.1 Virtualización de hardware

La virtualización de hardware, corresponde a administrar la parte física de un servidor o equipo que contiene a la aplicación que esta virtualizado y ser capaz de generar todos estos dispositivos físicos (discos, memoria, procesador, etc.) a un entorno virtualizado, [EPE2007], pudiendo así admitir por ejemplo varios equipos host; la única implicancia de este tipo de virtualización es que la memoria asignada a los equipos host es asignada de forma real así por ejemplo no se puede asignar más memoria a los equipos host de la que tiene el equipo físico.

1.7.2 Virtualización de sistemas operativos

Este tipo de virtualización va de la mano con la virtualización por hardware ya que el sistema operativo que se virtualiza se encuentra instalado sobre una plataforma virtualizada de hardware, en él se emulan sistemas operativos ya sea corriendo sobre el hypervisor o sobre un sistema operativo anfitrión.

Este tipo de virtualización, es muy ocupada en ambientes de desarrollo de software; por ejemplo para realizar pruebas de compatibilidad de la aplicación que se esté desarrollando, pudiendo así el programador testear en distintos ambientes (Linux, Windows, Mac) un mismo programa, logrando así obtener una amplia compatibilidad; de lo contrario para realizar estas pruebas el programador debería cambiar de escritorio y contar con otros equipos con los sistemas operativos a testear.

1.7.3 Virtualización de Aplicaciones

La virtualización de aplicaciones, tiene como finalidad, el lograr un manejo centralizado de aplicaciones que se encuentren disponibles en un servidor, [EPE2007], por ejemplo Citrix permite a los usuarios finales acceder a una aplicación determinada que está alojada en un servidor Citrix Xen, por medio de un navegador web, permitiendo así, que cualquier persona, en cualquier lugar y que cuente con un acceso a internet, pueda manejar la aplicación de su empresa como si estuviese en ella; además no requiere conexiones de seguridad anexas, como vpn(s), porque el servidor de Citrix encripta los datos para darle más seguridad a la conexión, además por ser administrada vía web puede ser ejecutada en cualquier dispositivo con una conexión a internet por ejemplo un celular, blackberry o dispositivos clam con acceso a internet.

VMware es otro de las empresas a la vanguardia en estos temas y tiene una contraparte para la virtualización de aplicaciones de Citrix se llama thinapp, el cual permite ejecutar una aplicación a través de un archivo de ejecución familiar, por ejemplo un archivo .exe o uno .msi permitiendo ejecutar la aplicación virtualizada sin tener ningún agente en el computador cliente.

1.7.4 Virtualización de equipos de redes

La virtualización de equipos va de la mano de la creciente necesidad de ahorro de espacio y ahorro de energía en las empresas, lo que enciende un paso a virtualizar equipos de redes permitiendo conectar equipos virtuales y físicos logrando así una distribución de los sistemas en la red de forma centralizada; además, de crear laboratorios de pruebas más flexibles para realizar testing de diseños de redes o planificación de estos.

Un ejemplo de este modelo de virtualización, es lo que hizo la empresa Cisco System con su equipo de red nexus 100v que es un switch virtual, para implementarlo por ejemplo en un servidor y realizar conexiones virtuales con otros equipos físicos como computadores, servidores o equipos de red,

existen también implementaciones open source, que nos permiten este tipo de virtualización como el Open vswitch que es el equipo virtual por defecto en aplicaciones como Xen cloud platform.

1.8 Problemas y posibles soluciones en la Virtualización

Problemas	Posible Solución.
Si se daña el disco duro se dañarán todas las máquinas virtuales que se encuentren en él.	Importantísimo el uso de RAID (ofrecen cierta tolerancia a fallos mediante la recuperación de información).
”Si se roban la máquina física, se pierden todas las máquinas virtuales.”	Realizar copias de seguridad de las imágenes de los Sistemas Operativos virtualizados y”Mejorar la seguridad”.
Cualquier problema que afecte al hardware, afectará a todas las máquinas virtuales (corriente, red...).	Utilización de sistemas redundantes (doble red, doble disco, doble fuente de corriente, etc.).

CAPITULO 2

VIRTUALIZACIÓN

2.1 Introducción

En este capítulo se describe la arquitectura del componente de virtualización xen 3.0, se da a conocer con más detalle las características de los dominios que se ejecutan en xen, así como las terminologías empleadas en la virtualización también se dará a conocer el hardware y software necesario para virtualizar huéspedes.

2.2 Que es Xen?

Xen, es una máquina virtual, permite ejecutar múltiples máquinas virtuales, cada una con su propio sistema operativo, y lo hace con un rendimiento prácticamente idéntico a una ejecución nativa de ese sistema. Normalmente se sufre una penalización del 2% y en casos extremos del 8%. [CD2007].

Para ejecutar un sistema operativo sobre Xen, este debe sufrir ligeras modificaciones para adaptarse a su API, pero una vez modificado el sistema operativo el resto de aplicaciones no debe sufrir ningún tipo de cambio, y esta es una de las grandes características de Xen, frente a otros paravirtualizadores como Denali, un gran rendimiento sin modificar aplicaciones y librerías. [CD2007].

Intel y AMD han anunciado que en sus nuevas gamas de procesadores con extensiones para virtualización soportarán Xen, haciendo así posible correr sistemas operativos sin modificar sobre Xen. Estas extensiones son conocidas como VanderPool y Pacifica para Intel y AMD respectivamente, aunque ya son muchos los sistemas operativos que poseen extensiones para poder correr sobre Xen, siendo algunos de ellos Linux, NetBSD, FreeBSD. [CD2007].

Windows XP corrió en las versiones de desarrollo de Xen pero no ha sido distribuido el parche por cuestiones de licencias de Windows y Microsoft. [CD2007].

XEN fue creado en el año 2003 en el laboratorio de computación de la Universidad de Cambridge, bajo lo que se conoce como el proyecto XEN Hypervisor liderado por Ian Pratt. Algunos de los miembros más destacados del proyecto son Keir Fraser, Steven Hand y Cristhian Limpach. Este mismo equipo fundo XENSource. [CD2007].

La herramienta XEN le permitirá crear y gestionar servidores virtuales dentro de un servidor físico. XEN está formado por los siguientes componentes:

- Hypervisor.
- Parche de kernel de Linux.
- Herramientas de administración por vía comando o gráfico.

El hypervisor es la pieza más importante dentro de XEN, es lo primero que inicia el GRUB de Linux ya que se encarga de controlar el hardware (CPU, memoria, etc.) y distribuir su uso entre las diversas máquinas virtuales. También permite desde XEN crear, controlar, monitorear y otras opciones a las máquinas virtuales. [CD2007].

Lo que también hace el hypervisor es que cada sistema operativo se maneja así mismo como si estuviera en un servidor real. El sistema base es comúnmente llamado Dom0 y a los servidores virtualizados se les conoce como DomU. [CD2007].

Dentro de los DomU existen dos tipos:

- HVM, lo que significa que el Sistema Operativo está siendo virtualizado por medio del módulo de virtualización del procesador de nuestra computadora.
- PVM, esta virtualización es por el método de SW, pero se tienen que modificar el S.O para que funcione.

Ventajas:

- Código abierto
- Proyecto maduro y probado
- XEN busca la manera de optimizar las capacidades de virtualización por HW en los procesadores de Intel y AMD.
- Buenas herramientas de administración/configuración completas y robustas.
- El mismo sistema de configuración de xen te permite crear tus propios scripts.
- Se pueden establecer varias configuraciones de red para las máquinas virtuales XEN.
- Migración de máquinas virtuales a otros equipos físicos.
- Documentación variada.

Desventajas:

- El soporte de instrucciones VT es relativamente reciente y existe poca documentación sobre este montaje, así como carece de cierta

funcionalidad que sí existe en la otra modalidad. Todo esto es cierto que es cuestión de tiempo.

- La administración solamente es realizada por el usuario root.
- Requiere parches en el kernel de GNU/Linux.

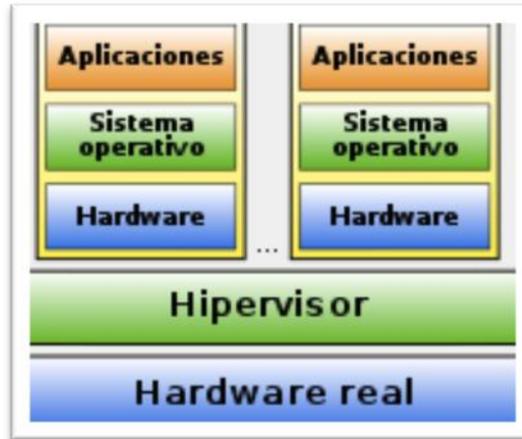
2.3 Que es el Hypervisor

Los hypervisores fueron originalmente desarrollados a principios de los años 70 para reducir costos, se consolidaban varias computadoras aisladas de diferentes departamentos de la empresa en una sola y más grande, el mainframe, capaz de servir a múltiples sectores. Al correr múltiples sistemas operativos a la vez, el hypervisor permite una consolidación dando robustez y estabilidad al sistema; aun si un sistema operativo colapsa, los otros continúan trabajando sin interrupción. [HYPD2011].

El Hypervisor proporciona garantías sobre los recursos a los servidores virtuales para asegurar que los niveles de servicio de cada aplicación se respeten, incluyendo CPU, memoria y entrada/salida.

Tipos.

Hypervisor tipo 1: También denominado nativo, unhosted o sobre el metal desnudo (bare metal), es software que se ejecuta directamente sobre el hardware, para ofrecer la funcionalidad descrita.



Hypervisor tipo 2: También denominado hosted, es software que se ejecuta sobre un sistema operativo para ofrecer la funcionalidad descrita.



2.4 Sistemas operativos soportados.

- Mandriva 2006 incluye Xen 2.6
- Novell's SUSE Linux Enterprise 10 incluye Xen 3.
- Fedora Core 5 incluye la versión 3 de Xen.

- Red Hat Enterprise Linux en su última versión 5, incluye soporte completo de Xen 3.
- Xenophilia es una distribución GNU/Linux que se basa en Xen.
- Xen demo CD es una ISO live CD basada en Debian que permite probar Xen en tu sistema sin instalarla en disco duro.
- Debian La versión 5.0 (lenny) incluye Xen 3.2, además de incluirlo en sus repositorios "testing" y "unstable".
- NetBSD 2.0 incluye soporte para Xen 1.2.
- Oracle VM Oracle basado en Xen
- CentOS 5

2.5 Hardware soportado

Actualmente soporta P6, eso quiere decir que soporta cualquier ordenador x86 fabricado después del cambio de siglo. También, en su versión 3.0, Xen soporta multiprocesadores y tecnología HT (HyperThreading). Esta soportada la arquitectura x86/64 y el port de IA64 está muy cerca de ser completado, y se espera que pronto podamos disfrutar de ports para PowerPC (IBM está trabajando en ello) y ARM. [RHI2007].

Xen no soporta ACPI o APM, por lo tanto funcionará pero no con todas las funcionalidades de un portátil, aunque los desarrolladores esperan poder soportar estas tecnologías de portátiles próximamente. [RHI2007].

Con el componente de virtualización, los anfitriones de 32-bit sólo pueden ejecutar huéspedes paravirtuales de 32-bit. Anfitriones de 64-bit sólo ejecutan huéspedes de 64-bits. Un anfitrión de 64-bit completamente virtualizado

puede ejecutar huéspedes de 32-bit, 32-bit PAE o 64-bit. Un anfitrión de 32-bit completamente virtualizado puede ejecutar huéspedes completamente virtualizados de 32-bit y 32-bit PAE (siglas en inglés de Physical Address Extension); para indicar al kernel que soporte pae se le tiene que especificar con el mandato mem. [RHI2010]:

```
title Red Hat Enterprise Linux Server (2.6.18-4.elxen) root (hd0, 0)
kernel /xen.gz-2.6.18-4-el5 mem=32G module /vmlinuz -2.6.18-
4.el5xen ro root=LABEL=/ module /initrd-2.6.18-4.el5xen.img
```

2.6 El dom0 y los domU

Dom0

Dom0, conocido también como hospedero o “host” del inglés, es el primer dominio (Sistema operativo) iniciado por el hypervisor xen. Cuenta con privilegios especiales, como ser capaz de crear e iniciar nuevos dominios, también es capaz de acceder al hardware directamente. Es responsable de ejecutar todos los controladores de dispositivo para el hardware, que se pondrá a disposición de otros dominios, como interfaces de red y discos, se ejecutará la BackendDriver, que multiplexa y envía a las solicitudes de hardware de la FrontendDriver en cada DomU, [XUM2008].

DomU

Un DomU, o también conocido como huéspedes o “guest” del inglés, es la contrapartida del Dom0, es un dominio sin privilegios, por defecto no tiene acceso al hardware. Se debe ejecutar un FrontendDriver para el hardware multiplexado que desea compartir con otros dominios. Un DomU es iniciado por el dom0, [XUM2008].

El núcleo de un DomU viene del sistema de archivos Dom0.

2.7 Requisitos de Instalación

Para Dom0 “Host”

- kernel-XEN
- XEN
- libvirt
- Linux con soporte de virtualización.
- Centos 5.2 o mayor, Fedora 8 o mayor, Debian, OpenSuse 10.3, etc.
- Procesador Pentium II o mayor.
- Para Full virtualización:

Procesadores Intel con virtual technology.

Intel-vt o amd-v

Para DomU “Guest”

- Para Virtualización

Cualquier tipo de Linux.

- Windows

Necesita de la tecnología intel-vt o amd-v “full virtualización”.

CAPITULO 3

XEN

3.1 Introducción

En el siguiente capítulo se verá la instalación de xen 3.0 bajo la plataforma Centos versión 5.5. A continuación se describirá la configuración de los Sistemas Operativos huéspedes en este caso un huésped paravirtual Centos 5.5, y fullvirtual Windows 7.

Como siguiente paso se elaborará una guía de administración para volúmenes lógicos y finalmente se describirá como realizar una migración en vivo de máquinas virtuales.

3.2 Instalación Xen.

- En la instalación de Xen, hay algunas opciones muy interesantes, como el manejo de volúmenes lógicos “LVM” y “RAID”, la finalidad de este capítulo, es instalar el Dom0 y los DomU en “LVM”, la cual representa mayor flexibilidad en comparación a las particiones estáticas.
- Para la instalación del hypervisor usar alguna distribución que tenga soporte de virtualización por ejemplo Centos 5.5, para esto se describe los pasos a seguir.



- Iniciar el boot desde el CD de instalación de Centos 5.5, luego esperar hasta que se muestre la pantalla de instalación, una vez iniciado el proceso de instalación presionar Enter.



- Una vez iniciado el proceso para la instalación del hypervisor nos pedirá comprobar si el disco de instalación tiene sectores defectuosos, seleccionar la opción de skip, para evitar la comprobación del CD, ya que el proceso de comprobación puede durar algunos minutos.



- En el caso de usar un disco nuevo, se mostrará una ventana con una advertencia, en la que indica que necesita crear una tabla de particionamiento, dar click en “Yes”, esto crea una nueva tabla de partición.



- Para crear una nueva tabla de partición seleccionar la opción de “Create custom Layout”, esta opción permite personalizar la tabla de particionamiento, acto seguido dar click en “Next”.



- Como siguiente paso se procede a crear una partición para el boot, para esto dar click en la opción de “New”, la cual nos permitirá añadir una nueva partición, se recomienda que la partición para boot sea ext3, primaria y con un tamaño de 100M, acto seguido dar click en el botón “Ok”.



- Procedemos a preparar una partición con soporte de “LVM”, para ello dar click en “New”, y seleccionar “physical volumen (LVM)” y la opción

“Fill to maximum allowable size”, esta opción crea una partición para el manejo de volúmenes lógicos, la cual es muy recomendado. Como siguiente paso, dar click en “Ok”.



- Ahora click en “LVM”, esto permite crear los logical Volume (LVM), cambiar el nombre del Volume Group Name por “dsk” (opcional), ahora presionamos en “Add”.



- Se procede a configurar las particiones del Dom0 con LV, para esto agregar una partición por cada directorio importante por ejemplo home (ext3), tmp (ext3), var (ext), swap (swap), root (ext3) con sus respectivos sistemas de archivos y puntos de montaje.



- Se recomienda que la sumatoria de las particiones del Dom0 no sobrepasen los 10G; para cada guest necesitamos ocupar una porción del volúmen lógico, ya que los domU son los servidores aislados que correrán independientemente, del Dom0.





- Una vez finalizada la creación de la estructura del hypervisor presionar en el botón “Next”.





- Como siguiente paso, presionar "Next", hasta llegar a la configuración de la red, seleccionar la opción "Manual configuration", y setear la ip con sus respectivos gateway y dns, esto con la finalidad de tener el Dom0 con ip estática.



- Como siguiente paso presionar en "siguiente", hasta llegar a la opción que permite instalar los servicios que tendrá el Dom0, se recomienda escoger simplemente la opción de virtualización, esto con el fin de optimizar recursos, pero el objetivo también es mostrar las herramientas gráficas, que faciliten la administración de los domU, para ello seleccionar un

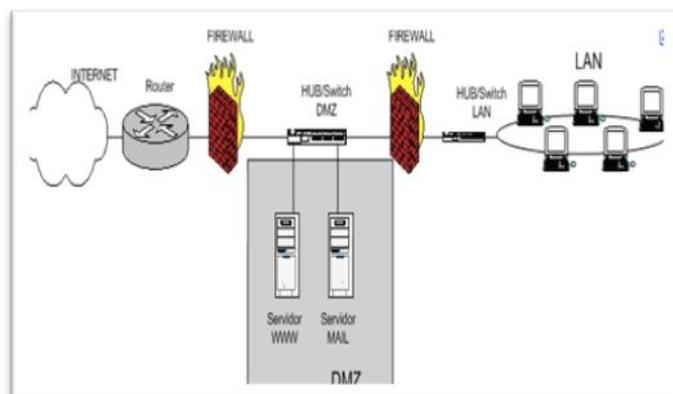
escritorio en el Dom0, en este caso seleccionar gnome o kde además del componente de virtualización.



- Una vez seleccionado los servicios que correrán en el hypervisor xen, dar click en “siguiente”, y comenzará el proceso de instalación, una vez finalizada la instalación proceder a reiniciar el servidor; acto seguido dar click en “reboot”.



- Una vez reiniciado el hypervisor se puede observar que el grub cuenta con un kernel Linux modificado con soporte de virtualización xen; presionamos “enter”, para continuar con la configuración.



- El siguiente paso es configurar algunas opciones, entre las más interesantes está la del firewall. Tomar en cuenta algunas consideraciones, siempre se recomienda colocar a los servidores públicos en una DMZ o zona perimetral; con el fin de aislar los servidores y proteger la red LAN, tener claro que los servidores brindan servicios al público y los firewall son las reglas, es decir que el hypervisor se trata de un servidor, entonces procedemos a deshabilitar las opciones de firewall “disabled” y selinux “disabled” ya que el objetivo no está en configurar reglas para el firewall y excepciones para SeLinux.



- Por último iniciar sesión. Una vez finalizada las configuraciones en el servidor deshabilitar al usuario root, esto por motivos de seguridad.
- Finalmente, ya tenemos instalado el servidor de virtualización, también se han instalado el hypervisor xen como dom0, el cual permite administrar Sistemas Operativos huéspedes o guest.

3.3 Configuración GRUB.

Proceder a configurar el gestor de arranque Linux. El kernel de virtualización Linux soporta 16 GB para sistemas de 32 bits y no más 64 GB para sistemas de 64 bits [MJ2010]. Se recomienda que se tenga al menos 256 MB por cada huésped que se está ejecutando en el sistema, para extender la memoria física o virtual se necesita de un componente llamado PAE (“siglas en inglés de Physical Address Extension”), ¿Cómo saber si contamos con este componente?

- Para determinar si un sistema soporta PAE introducir el siguiente comando en la terminal.

```
# cat /proc/cpuinfo | grep pae
```

```
flags: fpu tsc msr pae cx8 apic mtrr cmov pat clflush  
acpi mmx fxsr sse sse2 ss ht nx constant_tsc pni est  
ssse3 sse4_1
```

```
flags      : fpu tsc msr pae cx8 apic mtrr cmov pat  
clflush acpi mmx fxsr sse sse2 ss ht nx constant_tsc up  
pni est ssse3 sse4_1
```

- Se observan algunas características soportadas por el procesador entre ellas esta “pae”, esto quiere decir que modificando el grub.conf este puede reconocer hasta 16 GB de memoria RAM ya que se trata de un sistema de 32 bits.
- Los kernels de virtualización de Linux no se pueden ejecutar en sistemas sin PAE.
- Proceder a editar el archivo grub.conf; para ello introducir el siguiente comando en la terminal.

vi /boot/grub/grub.conf

- La información mostrada varía de acuerdo a las características y sistemas operativos instalados en el computador.



```
root@locanost-
File Edit View Terminal Tabs Help
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE: You have a /boot partition. This means that
# all kernel and initrd paths are relative to /boot/, eg.
# root (hd0,2)
# kernel /vmlinuz-version ro root=/dev/disk/root
# initrd /initrd-version.img
#boot=/dev/sda3
default=0
timeout=5
splashimage=(hd0,2)/grub/splash.xpm.gz
hiddenmenu
title CentOS (2.6.18-194.el5xen)
    root (hd0,2)
    kernel /xen.gz-2.6.18-194.el5
    module /vmlinuz-2.6.18-194.el5xen ro root=/dev/disk/root rhgb quiet
    module /initrd-2.6.18-194.el5xen.img
```

- Hay algunas opciones interesantes en este archivo, se comenzará describiendo cuales son:

default=0: Indica, que entrada, del fichero de configuración del grub, será selecciona por defecto, para ser iniciada, es decir, que sistema operativo de la lista que se muestra en el grub, se inicia por defecto.

timeout=5: Tiempo que se espera antes de iniciar el sistema operativo por defecto.

splashimage= (hd0, 2)/grub/splash.xpm.gz: Imagen que se muestra como fondo en el menú.

hiddenmenu: Se oculta el menú y solo se muestra la entrada por defecto antes de ser iniciada.

El inicio automático se puede cancelar y mostrar el menú completo.

ro: Se monta la partición en modo solo lectura para la carga.

root=/dev/dsk/root: Indica en donde se encuentra la raíz del sistema operativo.

kernel /xen.gz-2.6.18-194.el5: Es el kernel que se cargara al escoger la opción.

Algunas opciones que no se muestran en archivo y pueden resultar importantes en algún determinado caso.

mem = 32 G: Ayuda a extender nuestra memoria física o virtual.

password-md5: ayuda a proteger la entrada a un sistema operativo mediante una contraseña calculada con hash md5.

dom0_mem=800M: Restringir la memoria del dom0 a 800M.

dom0_max_vcpus: Este parámetro limita la cantidad de CPU visible para domain0.

- Un ejemplo de grub.conf modificado.



```
root@localhost:~# cat /etc/grub.conf
default=0
timeout=5
splashimage=(hd0,2)/grub/splash.xpm.gz
hiddenmenu
password --md5 $1$h31860$4u2T7eR1l.Qnts0Fc.#e1
title CentOS (2.6.18-194.el5xen)
    root (hd0,2)
    kernel /xen.gz-2.6.18-194.el5 mem=32GB
    module /vmlinuz-2.6.18-194.el5xen ro root=/dev/dsk/root rhgb quiet
    module /initrd-2.6.18-194.el5xen.img
```

- El archivo grub.conf, lo modificamos para que reconozca 32 GB en RAM, además de proteger el arranque mediante una contraseña generada con el comando.

grub-md5-crypt

El comando “grub-md5-crypt” nos pedirá ingresar una palabra y nos genera un hash en md5 de la palabra introducida.

3.4 Discos Virtuales.

En este paso se crearan los discos virtuales para los Sistemas Operativos huéspedes. Existen dos métodos para crear discos virtuales: el primero es mediante el mandato “dd” y el otro método es usando “lvm”.

Comando dd

Para crear discos duros virtuales por medio del comando “dd”, ingresar el siguiente comando en la terminal.

```
# dd if=/dev/zero of=/vm/server1.img oflag=direct bs=1M seek=4099
count=1
```

Opciones del comando dd:

OPCIÓN	DESCRIPCIÓN
if=/dev/zero	Lee desde la unidad infinita de ceros.
of=/vm/server1.img	Escribe en /vm/server1.img
oflag=direct	Escribe datos con comas.
bs=1M	Lee y escribe bytes a 1 MB.
seek=4099	Tamaño final del bloque.
count=1	Copia bloques de tamaño en bytes.



```
root@huiracocha images]# dd if=/dev/zero of=/var/lib/libvirt/images/server1.img
oflag=direct bs=1M seek=4099 count=1
1+0 records in
1+0 records out
1048576 bytes (1.0 MB) copied, 0.012181 seconds, 86.1 MB/s
you have new mail in /var/spool/mail/root
root@huiracocha images]# ll -h /var/lib/libvirt/images/server1.img
-rw-r--r-- 1 root root 4.1G May 18 00:10 /var/lib/libvirt/images/server1.img
```

Como se observa se ha preparado un disco virtual llamado “server.img” con un tamaño de 4 GB, el cual puede ser usado en la instalación de los huéspedes.

3.5 La herramienta kpartx

Cuando se trabaja con imágenes de máquinas virtuales, muchas veces hay la necesidad de acceder a los datos de imágenes sin ejecutarlo desde una máquina virtual, es decir, si hubiese una forma de montar toda la estructura de la imagen esto incluye particiones lvm etc.; para estas situaciones tenemos la herramienta kpartx.

Existen numerosos métodos para acceder a los datos desde archivos de imagen de huésped. Uno muy conocido, es el de utilizar la herramienta “kpartx”, descrita en esta sección, de modo de poder montar el sistema de archivos huésped como un dispositivo de bucle, al que puede accederse.

Los huéspedes deben estar desconectados antes que sus archivos puedan ser leídos. No es posible ni editar ni leer los archivos de un huésped activo, e intentarlo podría causar pérdida de datos o daños.

Para acceder a los datos de una imagen, proceder a verificar si el paquete kpartx está instalado en el sistema para ello introducir el siguiente comando.

```
# rpm -q kpartx
```

```
kpartx-0.4.7-42.el5_6.2
```

En el caso de no tener instalado la herramienta kpartx, proceder a instalarlo con el siguiente comando.

```
# yum install kpartx
```

Utilice `kpartx`, para conocer los mapeos de los dispositivos de particionamiento asociados a una imagen de almacenamiento basada en archivo. El siguiente ejemplo utiliza un archivo de imagen.

```
# kpartx -l /dev/dsk/kpartxExample
```

```
kpartxExample1 : 0 1975932 /dev/dsk/kpartxExample 63
```

```
kpartxExample2: 0 2152710 /dev/dsk/kpartxExample 1975995
```

`kpartxExample`, es un “lv” con dos particiones fijas; la primera partición es la partición de intercambio swap, y la segunda partición tiene una sistema de archivo ext3.

Para mapear las particiones `kpartxExample1` y `kpartxExample2`, ingresar el siguiente comando.

```
# kpartx -a /dev/dsk/kpartxExample
```

Una vez mapeado las particiones, verificar que funcione el mapeo de las mismas; deberían existir nuevos dispositivos en el directorio `/dev/mapper/`

```
# ls /dev/mapper/kpartxExample*
```

```
/dev/mapper/kpartxExample1 /dev/mapper/kpartxExample2
```

Ahora, la partición que se pretende montar es `kpartxExample2`, no interesa montar `kpartxExample1` ya que se trata de una partición de intercambio swap por ende para acceder a la partición `kpartxExample1`, introducimos el siguiente comando en una terminal.

```
# mount /dev/mapper/kpartxExample2 /mnt/ -o ro
```

```
# ls /mnt/
```

```
archivo1    archivo2    archivo3    archivo4    archivo5  
ArchivoEnKpartx  datos  lost+found
```

Los archivos ahora se encuentran disponibles para ser leídos en el directorio **/mnt/**. Léalos o cópielos.

Desmunte el dispositivo de manera que la imagen de huésped pueda ser reutilizada por él. Si el dispositivo se encuentra montado, el huésped no podrá acceder a la imagen, y por lo tanto, no podrá iniciarse.

```
# umount /mnt/
```

Desconecte el archivo de imagen de los mapeos de partición.

```
# kpartx -d /dev/dsk/kpartxExample
```

Ahora el huésped puede ser reiniciado.

```
# kpartx -a /dev/dsk/kpartxExample
```

Cómo acceder a los datos desde huéspedes ejecutados en un entorno de manejo de volúmenes lógicos “lvm”.

Numerosos huéspedes Linux utilizan volúmenes de “Logical Volume Management (LVM)”. Se necesitan algunos pasos adicionales para leer datos de volúmenes LVM de imágenes de almacenamiento virtual.

Agregar los mapeos de partición para el serverMail a los dispositivos reconocidos en el directorio `/dev/mapper/`.

```
# kpartx -a /dev/dsk/serverMail
```

En el ejemplo, los volúmenes LVM se encuentran en una segunda partición. Los volúmenes necesitan de otro análisis con el comando `vgscan`, para que pueda encontrar los nuevos grupos de volúmenes.

```
# vgscan
```

```
Reading all physical volumes. This may take a while...
```

```
Found volume group "VolGroup00" using metadata type lvm2
```

```
Found volume group "dsk" using metadata type lvm2
```

Active el grupo de volúmenes en la partición (denominada `VolGroup00` por defecto), con el comando `vgchange -ay`.

```
# vgchange -ay VolGroup00
```

```
2 logical volumes in volume group VolGroup00 now active.
```

Utilice el comando `lvs` para observar información relacionada con los nuevos volúmenes. Los nombres de estos volúmenes (la columna `LV`) son necesarios para poder montarlos.

```
# lvs
```

```
LV VG Attr LSize Origin Snap% Move Log Copy% Convert
```

```
LogVol00 VolGroup00 -wi-a- 9.06G
```

```
LogVol01 VolGroup00 -wi-a- 608.00M
```

```
kpartxExample dsk -wi-a- 1.97G
```

```
root      dsk      -wi-ao 19.53G
```

```
serverMail dsk      -wi-ao 9.78G
```

```
serverWeb  dsk      -wi-a- 9.78G
```

```
swap      dsk      -wi-ao 1.00G
```

Monte el volúmen /dev/VolGroup00/LogVol00 en el directorio /mnt/

```
# mount /dev/VolGroup00/LogVol00 /mnt/
```

Verificamos que la partición se agregó al directorio /mnt

```
# ls /mnt/
```

```
bin boot dev etc home lib lost+found media misc mnt
net opt proc root sbin selinux srv sys tmp usr var
```

Ahora los archivos se encuentran disponibles para ser leídos en el directorio /mnt/. Léalos o cópielos.

Desmonte el dispositivo de manera que la imagen de huésped pueda ser reutilizada por él. Si el dispositivo se encuentra montado, el huésped no podrá acceder a la imagen, y por lo tanto, no podrá iniciarse.

```
# umount /mnt/
```

Desconecte el grupo de volúmenes VolGroup00

```
# vgchange -an VolGroup00
```

0 logical volume(s) in volume group "VolGroup00" now active

Desconecte el archivo de imagen de los mapeos de partición.

```
# kpartx -d /var/lib/libvirt/images/guest1.img
```

Ahora el huésped puede ser reiniciado.

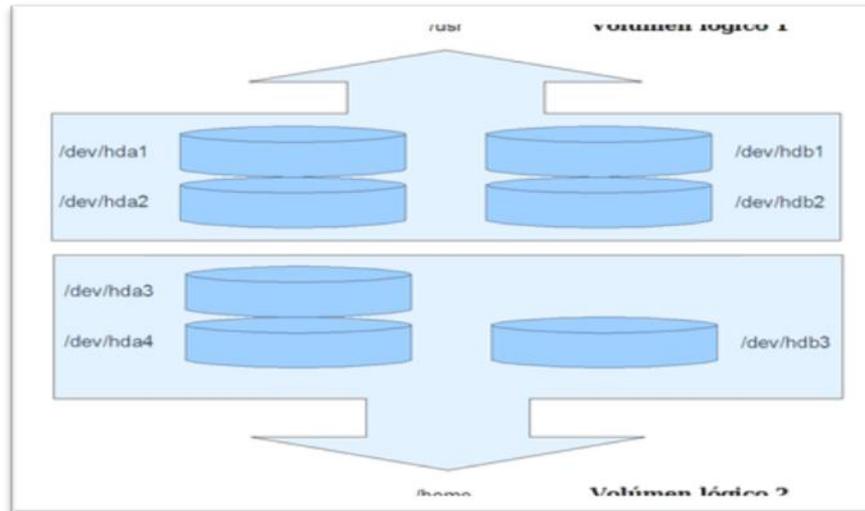
3.6 Configuración LVM y Raid.

LVM es una implementación de un administrador de volúmenes lógicos para el kernel Linux. Se escribió originalmente en 1998 por Heinz Mauelshagen, que se basó en el administrador de volúmenes de Veritas usado en sistemas HP-UX.

LVM incluye muchas de las características que se esperan de un administrador de volúmenes, incluyendo:

- Redimensionado de grupos lógicos
- Redimensionado de volúmenes lógicos
- Instantáneas de sólo lectura (LVM2 ofrece lectura y escritura)
- RAID0 de volúmenes lógicos.

LVM no implementa RAID1 o RAID5, por lo que se recomienda usar software específico de RAID para estas operaciones, teniendo las LV por encima del RAID



En la imagen se observa cómo trabaja LVM. Los volúmenes lógicos agrupan particiones físicas de disco, y éstos a su vez, aunque no está representado en la figura, se engloban en un grupo lógico. De esta forma, /home se compone de hda3, hda4 y hdb3, y a su vez, /usr engloba a hda1, hda2, hdb1 y hdb2.

La gestión de volúmenes lógicos proporciona una vista de alto nivel sobre el almacenamiento en un ordenador, en vez de la tradicional vista de discos y particiones, algunas características de los LVM.

- Los volúmenes de almacenamiento bajo el control de LVM pueden ser redimensionados y movidos a voluntad, aunque esto quizá necesite actualizar las herramientas del sistema.
- LVM también permite la administración en grupos definidos por el usuario, permitiendo al administrador del sistema tratar con volúmenes llamados, por ejemplo, "ventas" o "desarrollo", en vez de nombres de dispositivos físicos, como "sda" o "sdb"

Una de las decisiones que afronta un usuario instalando GNU/Linux es cómo particionar el disco duro. La necesidad de estimar cuánto espacio será necesario para el sistema, para los temporales o para los datos personales, puede convertirse en algo problemático, por lo que muchos usuarios optan

por crear una partición que ocupe todo el disco y allí introducir los datos. Aun habiendo estimado correctamente cuanto espacio se necesita para /home, /usr, /tmp, o cualquier otro directorio importante, es bastante común que nos quedemos sin espacio en estas particiones, cuando tenemos espacio de sobra en alguna otra.

Ventajas de usar LVM en un sistema pequeño

- Con el uso de un administrador de volúmenes lógicos, el disco completo puede ser asignado a un único grupo lógico y definir distintos volúmenes lógicos para almacenar /home u otros directorios. En el caso que nos quedemos sin espacio, por ejemplo, en /home, y tenemos espacio en /opt, podríamos redimensionar /home y /opt y usar el espacio que le hemos quitado a /opt y añadirselo a /home. Hay que tener en cuenta, que para realizar esto, el sistema de ficheros debe soportar el redimensionado por arriba y por abajo, como ReiserFS.
- Otra alternativa sería dejar una cierta cantidad de espacio del disco sin asignar y cuando fuera necesario, expandir el volumen.

Ventajas de usar LVM en un sistema grande

- Administrar un sistema con muchos discos es un trabajo que consume tiempo, y se hace particularmente complejo si el sistema contiene discos de distintos tamaños. Balancear los requerimientos de almacenamiento de distintos usuarios (a menudo conflictivos) puede ser una tarea muy laboriosa y compleja.
- Los grupos de usuarios (llamémosles por ejemplo administración, desarrollo, etc.) pueden tener sus volúmenes lógicos y éstos pueden crecer lo que sea necesario, y el administrador puede realizar las operaciones oportunas sobre dichos volúmenes.

Cuando un nuevo disco se añade al sistema, no es necesario mover los datos de los usuarios. Simplemente se añade el nuevo disco al grupo lógico correspondiente y se expanden los volúmenes lógicos todo lo que se considere adecuado. También se pueden migrar los datos de discos antiguos a otros nuevos, de forma totalmente transparente al usuario.

Anatomía de un LVM

Un LVM se descompone en tres partes:

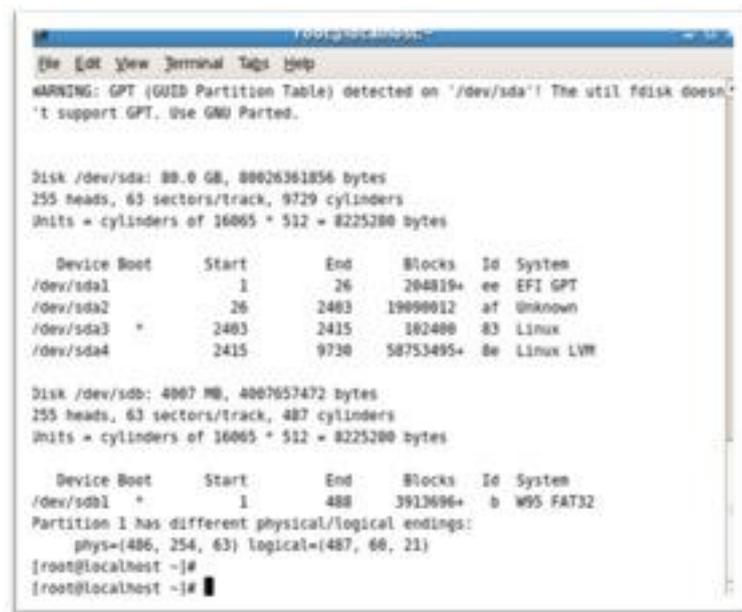
- **Volúmenes físicos (PV):** Son las particiones del disco duro con sistema de archivos LVM. (raid's)
- **Volúmenes lógicos (LV):** es el equivalente a una partición en un sistema tradicional. El LV es visible como un dispositivo estándar de bloques, por lo que puede contener un sistema de archivos (por ejemplo /home)
- **Grupos de volúmenes (VG):** es la parte superior de la LVM. Es la "caja" en la que se hospedan los volúmenes lógicos (LV) y volúmenes físicos (PV). Se puede ver como una unidad administrativa en la que se engloban los recursos. Hay que hacer notar que mientras un PV no se añada al VG, no puede comenzar a usarlo.

Para mostrar como configurar y administrar volúmenes lógicos realizar las siguientes tareas; instalar dos discos en el computador, usar el segundo disco para pruebas con lvm y raid, una vez contado con los recursos proceder con la administración de volúmenes lógicos.

- 1. Crear un PV (Volúmen Físico).

Para crear una pv primero averiguar cómo se encuentra la tabla de particiones además de los discos duros disponibles, introducir el siguiente comando en una terminal.

```
# fdisk -l
```



```
WARNING: GPT (GUID Partition Table) detected on '/dev/sda'! The util fdisk doesn't support GPT. Use GNU Parted.

Disk /dev/sda: 88.0 GB, 8826361856 bytes
255 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            1           26    204819+   ee  EFI GPT
/dev/sda2            26          2403    1999012   af  Unknown
/dev/sda3            *        2403         2415     102400   83  Linux
/dev/sda4            2415         9730    58753495+  8e  Linux LVM

Disk /dev/sdb: 4007 MB, 4007657472 bytes
255 heads, 63 sectors/track, 487 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            *            1           488    3913696+   b  W95 FAT32
Partition 1 has different physical/logical endings:
   phys=(486, 254, 63) logical=(487, 60, 21)
[root@localhost ~]#
[root@localhost ~]#
```

Como se puede observar hay dos unidades; la primera unidad denominado /dev/sda con cuatro particiones, el cual contiene el hypervisor xen, implementado con LVM; una segunda unidad /dev/sdb con una sola partición.

Una vez analizado la tabla de particiones procedemos a preparar un disco con soporte para “lvm”, con el fin de no dañar ni modificar el LVM que se encuentra en /dev/sda, usar el segundo disco que en realidad se trata de una memoria usb, el cual se usa por motivos de prueba, el procedimiento para crear modificar y eliminar lvm en un disco duro real, es el mismo.

- Eliminar la tabla de partición que se encuentre instalada en la unidad, para esto ejecutar el siguiente comando.

```
# dd if=/dev/zero of=/dev/sdb 'Este paso lo podemos omitir.
```

- El comando dd ayuda entre otras cosas a formatear la unidad, su sintaxis es la siguiente “dd if=unidad de origen, of=unidad de destino”.
- Hay que tener mucho cuidado con el comando “dd if=/dev/zero of=/dev/sdb” ya que muchas veces ha sido responsable de pérdidas de información imposible de recuperarlos ya que se trata de un formateo a bajo nivel.
- Una vez preparado el dispositivo procedemos a instalar un sistema de archivo con soporte de lvm. ejecutar los siguientes comandos en el mismo orden.

```
# fdisk /dev/sdb
```

```
h #Muestra las opciones de fdisk.
```

```
n #Crea una nueva partición.
```

```
p #Indica que es partición primaria.
```

```
1 #Indica el número de la partición.
```

```
1 #Primer cilindro.
```

```
+1024M #El tamaño de la partición es de 1024(1GB)
```

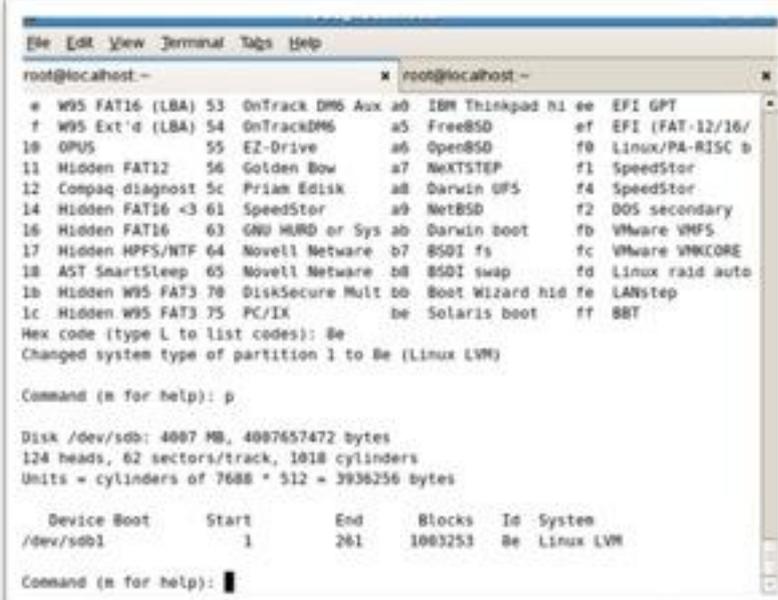
```
t #Permite preparar la partición para un
#sistema en especial en este caso un LVM.
```

```
8e #Es el código para LVM
```

p

#Imprime la tabla de partición.

La opción “p” del comando “fdisk” nos sirve para imprimir la tabla de partición actual, destacando el tipo de sistema de archivo.



```
root@localhost:~# fdisk -l
Disk /dev/sdb: 4007 MB, 4007657472 bytes
124 heads, 62 sectors/track, 1018 cylinders
Units = cylinders of 7680 * 512 = 3932160 bytes

Device Boot      Start         End      Blocks   Id  System
/dev/sdb1        1           261     1003253   Be  Linux LVM
```

Una vez creado y preparado la primera partición con soporte lvm, proceder a crear una segunda partición con lvm.

Para crear una segunda partición con soporte lvm realizar los mismos pasos del primer lvm, para ello introducir las siguientes opciones en el mismo orden.

n

#Nueva Partición.

p

#Primaria

2

#Numero de partición.

Dar Enter

#Cilindro de inicio.

+1024M #Tamaño de la partición.

T #Preparar la partición para LVM.

2 #Numero de partición a preparar.

8e #Código para LVM.

P #Imprime la tabla de partición.

w #Guardar y salir.

Una vez finalizada la preparación de las unidades procedemos a revisar la tabla de partición escribiendo el siguiente comando.

```

root@localhost ~
[root@localhost ~]# fdisk -l

WARNING: GPT (GUID Partition Table) detected on '/dev/sda'! The util fdisk doesn't support GPT. Use GNU Parted.

Disk /dev/sda: 80.0 GB, 80026361856 bytes
255 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            1           26     204819+   ee  EFI GPT
/dev/sda2            26          2403    19090012   af  Unknown
/dev/sda3            *          2403          2415     102400   83  Linux
/dev/sda4            2415         9730    50753495+   8e  Linux LVM

Disk /dev/sdb: 4007 MB, 4007657472 bytes
124 heads, 62 sectors/track, 1018 cylinders
Units = cylinders of 7680 * 512 = 3936256 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            1           261     1003253   8e  Linux LVM
/dev/sdb2            262          522     1003284   8e  Linux LVM
root@localhost ~#

```

Finalmente se ha logrado obtener dos particiones preparados con LVM, dichas particiones simulan a dos discos diferentes, cada disco o partición está preparada para usar LVM, procedemos con la creación de volúmenes físicos.

Con la partición /dev/sdb1 crear la primera PV para esto ejecutar los siguientes comandos en una terminal.



```
root@localhost ~
[root@localhost ~]# pvcrcate /dev/sdb1
Physical volume "/dev/sdb1" successfully created
[root@localhost ~]# pvcrcate /dev/sdb2
Physical volume "/dev/sdb2" successfully created
```

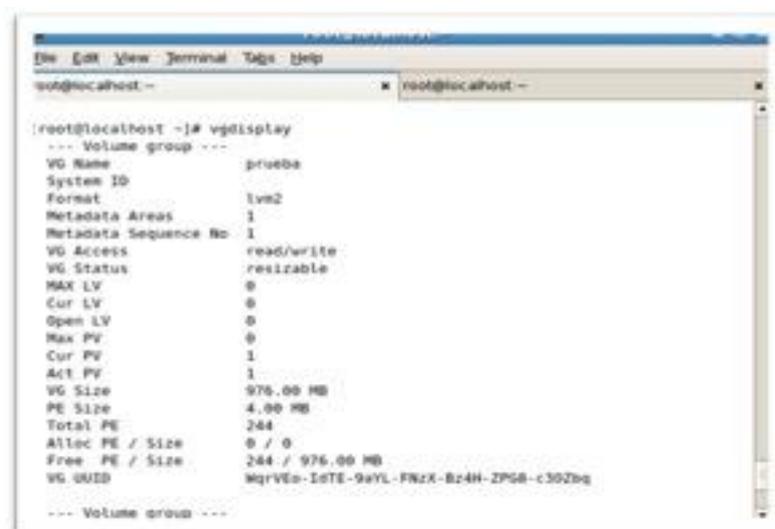
Una vez finalizada la creación de la “pv”, el siguiente paso es crear el “vg” (Grupo de Volúmen).

Para crear una VG introducir el siguiente comando en una terminal.

```
# vgcreate prueba /dev/sdb1
```

Con el comando vgcreate lo que hace es crear un vg con el nombre de “prueba”, inicialmente con una sola unidad /dev/sdb1, el motivo por el cual se lo realiza de esa manera es para demostrar las ventajas que se dan al usar LVM.

Hasta el momento se ha creado un vg con un tamaño de 1024M, para comprobar esto se puede ejecutar el siguiente comando en un terminal.



```
root@localhost ~
[root@localhost ~]# vgdisplay
--- Volume group ---
VG Name                prueba
System ID
Format                 lvm2
Metadata Areas         1
Metadata Sequence No  1
VG Access               read/write
VG Status               resizable
MAX LV                 0
Cur LV                 0
Open LV                 0
Max PV                 0
Cur PV                 1
Act PV                  1
VG Size                976.00 MB
PE Size                 4.00 MB
Total PE                244
Alloc PE / Size        0 / 0
Free PE / Size          244 / 976.00 MB
VG UUID                 MqrVGo-IdTE-9aYL-FNrx-Bz4H-ZP68-c30Zbq
--- Volume group ---
```

Una vez ejecutado el comando “vgdisplay”, se puede observar el nombre del vg, el formato, el VG Size, etc.

Una vez finalizada la creación del “vg”, se procede a crear los “lv” para ello ejecutar los siguientes comandos en un terminal.

```
# lvcreate -ntmp -L512M prueba
```

```
# mkfs.ext3 /dev/prueba/tmp
```

El primer comando “lvcreate” permite crear el “lv”, con la opción -n se permite escribir un nombre para el “lv” y con la opción -L se puede fijar el tamaño del “lv” y por último se debe indicar a que “vg” pertenece, en este caso a “prueba”.

Una vez finalizada la creación del lv, proceder a crear un filesystem para el nuevo “lv”, en este caso escoger ext3.

Como siguiente paso, crear un segundo “lv” que simule al directorio home, para esto escribir los siguientes comandos en una terminal.

```
# lvcreate -nhome -L200M prueba
```

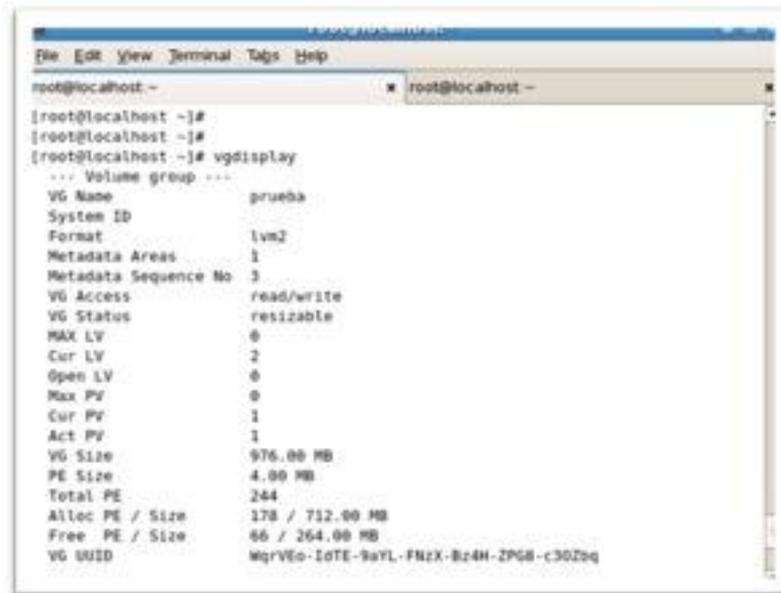
```
# mkfs.ext3 /dev/prueba/home
```

Una vez finalizada la creación del segundo “lv”, creamos un tercer “lv” con un tamaño de 512M que simule al directorio var, escribir lo siguiente en una terminal.

```
# lvcreate -nvar -L512M prueba
```

```
Insufficient free extents (66) in volume group prueba: 128 required
```

Se puede observar un mensaje que indica que el “vg” se quedó sin espacio, introducir el siguiente comando para comprobar el espacio del vg.



```
root@localhost ~
root@localhost ~
root@localhost ~
root@localhost ~# vgdisplay
... Volume group ...
VG Name                prueba
System ID
Format                 lvm2
Metadata Areas        1
Metadata Sequence No  3
VG Access              read/write
VG Status              resizable
MAX LV                 0
Cur LV                2
Open LV                0
Max PV                 0
Cur PV                1
Act PV                 1
VG Size                976.00 MB
PE Size                4.00 MB
Total PE               244
Alloc PE / Size       178 / 712.00 MB
Free PE / Size         66 / 264.00 MB
VG UUID                WqrVGo-IdTE-9aYL-FNzX-Bz4H-ZPG8-c30Zdq
```

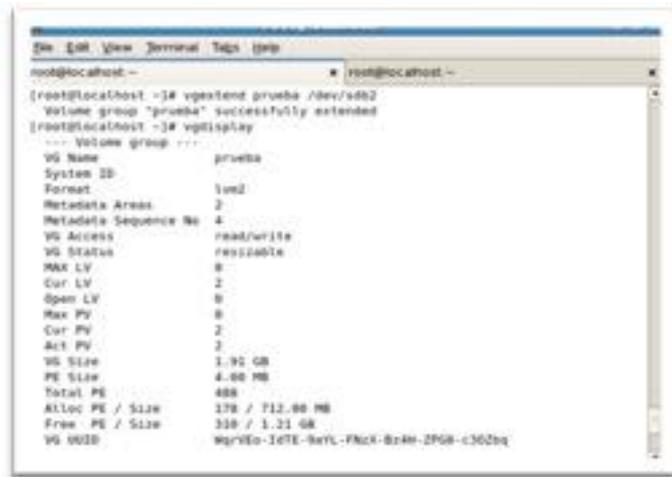
Una vez desplegada la información del “vg” procedemos a observar el “Free PE/size 66/ 264.00 MB”, esto indica que el espacio del “vg” ahora es de 264MB, esto es lógico ya que se ha creado dos “lv”; uno con 512 MB; otro con 200 MB más 264 MB suman 976 MB y si le sumamos los 48 MB reservados por el “vg” dan como resultado los 1024 MB, Se pretende añadir una nueva “lv” con un tamaño de 512 MB este sobrepasa la capacidad del “vg”, para solucionar este problema proceder a incrementar el tamaño del “vg” introducir el siguiente comando en una terminal.

```
# vgextend prueba /dev/sdb2
```

El segundo disco se ha reservado con la finalidad de conocer las ventajas al usar “lvm” como se puede observar una de las ventajas se da al redimensionar el “vg”, permitiendo incrementar o decrementar el tamaño sin mayor problema.

Una vez incrementado el tamaño del “vg” comprobamos el nuevo tamaño introduciendo el siguiente comando en una terminal.

```
# vgdisplay
```



```
root@localhost ~# vgsentent prueba /dev/sdb2
Volume group "prueba" successfully extended
root@localhost ~# vgdisplay
--- Volume group ---
VG Name                prueba
System ID              lvm2
Format                 lvm2
Metadata Areas        2
Metadata Sequence No  4
VG Access              read/write
VG Status              resizable
MAX LV                8
Cur LV               2
Open LV              0
Max PV               8
Cur PV              2
Act PV              2
VG Size              1.91 GB
PE Size              4.00 MB
Total PE             488
Alloc PE / Size     178 / 712.00 MB
Free PE / Size      310 / 1.21 GB
VG UUID              WqrVto-1c7E-8arL-FNck-8r4H-2F08-c362bq
```

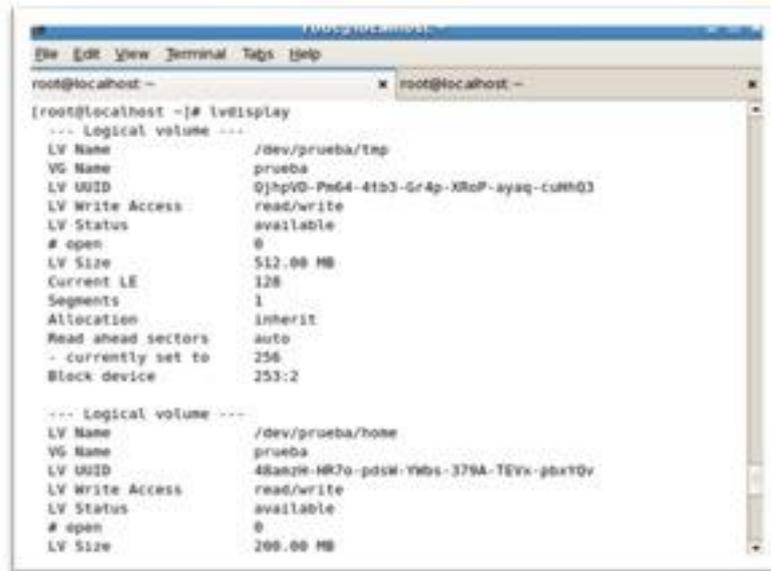
Se puede observar que el tamaño del “vg” prueba, incrementó su tamaño a 1.91 GB.

Una vez solucionado el problema del espacio se procede a crear el tercer “lv”, escribimos el siguiente comando en una terminal.

```
# lvcreate -nvar -L500M prueba
```

```
# mkfs.ext3 /dev/prueba/var
```

Finalizado la creación del tercer “lv” se puede observar el resultado de los “lv’s”, introducir el siguiente comando en una terminal.



```
root@localhost ~# lvs
--- Logical volume ---
LV Name                /dev/prueba/tmp
VG Name                 prueba
LV UUID                0jhpVD-Pm64-4tb3-Gr4g-XRoP-ayaq-cuH03
LV Write Access        read/write
LV Status               available
# open                  0
LV Size                 512.00 MB
Current LE              128
Segments                1
Allocation              inherit
Read ahead sectors     auto
 - currently set to    256
Block device            253:2

--- Logical volume ---
LV Name                /dev/prueba/home
VG Name                 prueba
LV UUID                48aayE-HR7o-pdsw-YHbs-379A-TEVx-gbrtQv
LV Write Access        read/write
LV Status               available
# open                  0
LV Size                 200.00 MB
```

Se puede observar los “lv” creados así como sus respectivos tamaños.

Como siguiente caso, simular que home se quedó sin espacio, por ende se pretende incrementar el tamaño de home, esto es posible gracias al uso de las “lvm”, se tiene disponible unos 700 MB aproximadamente en el “vg”. Entonces se pretende incrementar a home de 200MB a 900MB, para ello introducir los siguientes comandos en una terminal.

```
# e2fsck -f /dev/prueba/home
```

```
# lvresize /dev/prueba/home -L900M
```

```
# resize2fs /dev/prueba/home
```

El comando e2fsck comprueba los posibles defectos en el sistema de archivos y si es posible los corrige, lvresize es el mandato para redimensionar el “lv”, luego al aumentar el “lv” también se debe aumentar el filesystem, para ello se usa resize2fs.

Como resultado final, ahora la partición para home tiene el tamaño de 900 MB, se procede a comprobar el tamaño del “lv”.

```
root@localhost:~  
df  
Filesystem            Size  Used Avail Use% Mounted on  
--- Logical volume ---  
/dev/prueba/home     900M  40M  860M   5% /  
--- Logical volume ---  
/dev/prueba/var      500M  100M 400M  20% /var
```

Se tiene dos “vg’s” uno llamado “prueba” y el otro llamado “dsk”, por el momento se ha usado el vg de prueba. El vg de prueba tiene un tamaño de 1.91 GB con 40 MB libre, dentro del “vg” se tiene tres “lv’s” con los siguientes tamaños: tmp con 512 MB, home con 900 MB, var con 500 MB.

Como siguiente paso se procede a incrementar el tamaño de “home”, una vez más, se pretende pasar de 900 MB a 1300 MB, pero la vg cuenta con 40 MB libre, qué hacer?, en este caso, se tiene que decidir si un “lv” tiene un tamaño más grande de lo que necesita, optar por redimensionarlo, por ende se supone que tmp tiene un tamaño más grande de lo necesario, entonces el paso a seguir es reducir tmp de 512 MB a 100 MB, con eso es más que suficiente para incrementar home a 1300 MB.

Para lograr el objetivo, “disminuir el tamaño de tmp y aumentar el tamaño de home”, se procede a realizar la tarea por pasos. Como primer paso, disminuir el tamaño de tmp, introducir los siguientes comandos en una terminal.

```
# e2fsck -f /dev/prueba/tmp
```

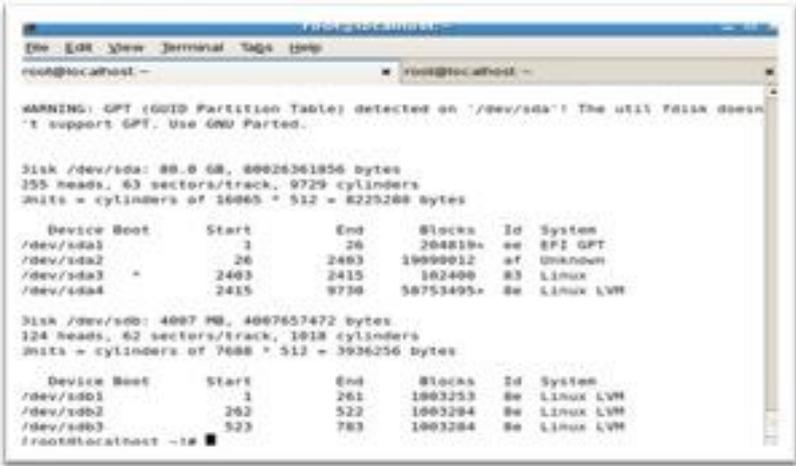
```
# resize2fs /dev/prueba/tmp 100M
```

```
# lvresize /dev/prueba/tmp -L 100M
```

Una vez introducidos los comandos proceder a entender los pasos realizados, en primer lugar se verifica que no haya errores en la “lv”, para luego reducir el filesystem y por ultimo reducimos la “lv”, el proceso fue un poco diferente que el de aumentar el tamaño del “lv”, en la reducción del “lv”, primero se reduce el filesystem y luego el “lv”, esto es muy importante, ya que al no tener presente esta consideración, se puede dar el caso de pérdida de información, de echo volaría una parte del filesystem.

Como siguiente punto se pretende cambiar un disco defectuoso, para esto se necesita otro disco con el fin de remplazarlo por el disco defectuoso, una vez obtenido un disco nuevo /dev/sda3, que se lo ha preparado con LVM y que el disco defectuoso es /dev/sda1, procedemos a preparar el disco defectuoso para ser retirado.

Ya instalado el nuevo disco, comprobar el número de discos actualmente instalados en el sistema operativo.



```
root@localhost:~# fdisk -l

WARNING: GPT (GUID Partition Table) detected on '/dev/sda': The util fdisk doesn't support GPT. Use GNU Parted.

Disk /dev/sda: 80.0 GB, 80026368000 bytes
355 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1            1           26       204816   ee   EFI GPT
/dev/sda2            26          243       1999012   af   Unknown
/dev/sda3            *          243          245         162400   83   Linux
/dev/sda4            245          9730      58753495   8e   Linux LVM

Disk /dev/sdb: 4007 MB, 4007657472 bytes
124 heads, 62 sectors/track, 1918 cylinders
Units = cylinders of 7680 * 512 = 3936256 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1            1           261       1993253   8e   Linux LVM
/dev/sdb2            262          522       1993284   8e   Linux LVM
/dev/sdb3            523          783       1993284   8e   Linux LVM

root@localhost:~#
```

El siguiente paso es remover el disco defectuoso /dev/sda1, para ello introducir los siguientes comandos en una terminal.

```
# vgextend prueba /dev/sda3
```

```
# pvmove /dev/sdb1
```

```
# vgreduce prueba /dev/sdb1
```

```
# pvremove /dev/sdb1
```

El comando `vgextend` sirve para incrementar el tamaño del `vg`, para luego mover toda la información de `/dev/sdb1` hacia los otros discos con `pvmove`, luego reducir el `vg` con `vgreduce` y por último remover el PV defectuoso.

3.7 Administración de máquinas virtuales con `xm`

La herramienta de virtualización `xen` brinda una serie de herramientas para facilitar la administración de máquinas virtuales, procedemos a crear los `guest`, El primer sistema operativo en cargar es el `dom0` y viene instalado por defecto. El `dom0` es el encargado de administrar los `guest` o huéspedes ofreciendo recursos cuando lo necesiten.

La herramienta de virtualización `xen`, nos ofrece un comando nativo para administrar los `guest`, el mandato `xm`, para mostrar lo interesante de la herramienta `xen`, procedemos primero a crear el `guest`.

El primer `guest` a crear se lo llamará `serverMail`, que será un sistema operativo CentOS paravirtual.

Para ello se instalará cada `guest` en un `“LV”`, con esto se aprovecha las ventajas que tiene el uso de los LVM, ingresar los siguientes comandos en un terminal.

```
# lvcreate -nserverMail -L10000M dsk
```

```
# mkfs.ext3 /dev/dsk/serverMail
```

Como se puede observar, ahora se usa el `“vg”` `dsk` y creamos el `“lv”` `serverMail` con un tamaño de 10000 MB.

CentOS 5 ofrece una herramienta llamada virt-install, que hace una serie de preguntas y permite instalar una máquina paravirtual o full virtual.

3.7.1 Instalación de un huésped CentOS 5.5 con el método paravirtual

En la instalación de un huésped paravirtual se usará el Sistema Operativo Centos 5.5, dado que Centos 5.5 entra en la categoría de un sistema operativo que permite modificar su kernel para fusionarlo con el hypervisor.

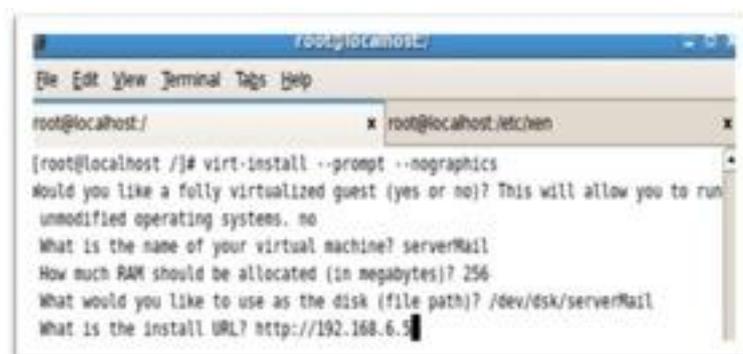
Se pretende instalar Centos 5.5 como huésped mediante la herramienta virt-install y usarlo como servidor de mails. Para instalar un guest con virt-install se realiza los siguientes pasos.

Introducir el cd de instalación y montar el disco en /var/www/html para ello ejecutar los siguientes comandos en una terminal

```
# mount /dev/cdrom /var/www/html
```

```
# service httpd start
```

Hasta ahora lo que se ha hecho es introducir el cd de instalación y montarlo en /var/www/html, para luego iniciar el servidor web con el fin de conseguir un repositorio local, con los paquetes de instalación que serán usadas por el Guest, ahora proceder a ejecutar el comando virt-install de la siguiente manera.



```
root@localhost:~# virt-install --prompt --nographics
Would you like a fully virtualized guest (yes or no)? This will allow you to run
unmodified operating systems. no
What is the name of your virtual machine? serverMail
How much RAM should be allocated (in megabytes)? 256
What would you like to use as the disk (file path)? /dev/dsk/serverMail
What is the install URL? http://192.168.6.5
```

Cuando se introduce el comando `virt-install`, se realiza una serie de preguntas entre ellas está el método a usar en la instalación del Guest. Como saber si un procesador tiene soporte de virtualización (full virtualización)? Para saber si el procesador soporta full virtualización se lo verifica en el archivo virtual `cpuinfo`.

```
# cat /proc/cpuinfo | grep "svm" #en el caso de un procesador amd
```

```
# cat /proc/cpuinfo | grep "vmx" #en el caso de un procesador intel
```

En el archivo `cpuinfo` se extrae muchas características del procesador, si el procesador en el apartado `flags` cuenta con soporte de virtualización, se podría instalar un guest en modo full-virtualización, se escogerá el modo `para-virtual`, también `"virt-install"` pregunta por el nombre del guest, escribir `serverMail`, luego se pregunta por el disco virtual en donde se instalará el huésped, se ingresa el path del `"lv"` `serverMail`, que fue creado con anterioridad, por último se ingresa la URL donde se encuentra el servidor web, se recomienda no usar `127.0.0.1` ya que muchas veces el hypervisor no encuentra esa URL.

Una vez configurado la URL del repositorio procedemos con la configuración de la dirección ip del servidor mail, los datos introducidos puede variar dependiente a la red en la que se encuentre el servidor.

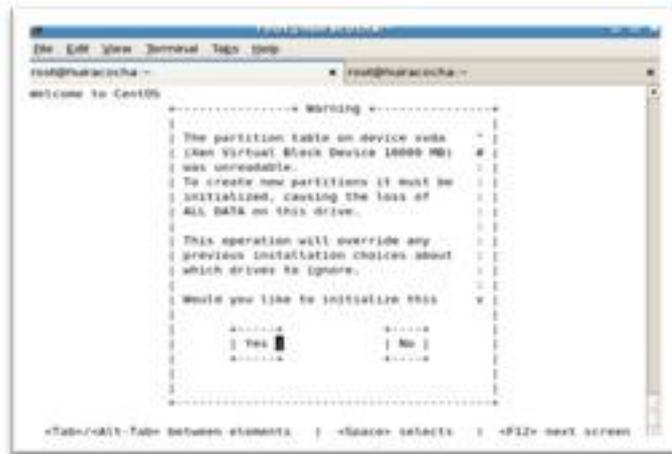


Una vez configurado la dirección “ip” el “dns” y la puerta de “enlace”, escoger “ACEPTAR” y esperar hasta que determine el nombre del host y el dominio.



Una vez resuelto el nombre del host y el dominio al que pertenece, se tiene que esperar hasta que el proceso de instalación encuentre los archivos necesarios para la instalación del huésped, que en este caso se encuentra en un repositorio web local.

tabla da partición; para crear una nueva tabla de partición escoger la opción “yes” y proceder a dar enter.



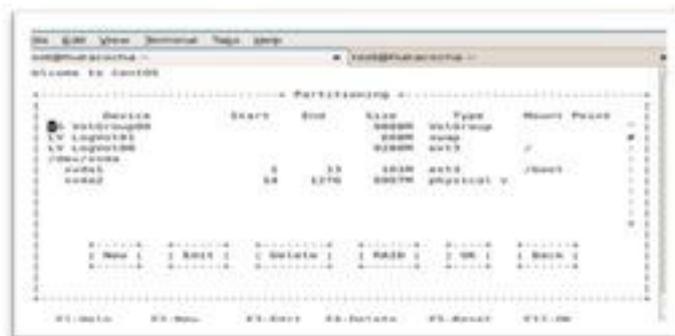
El siguiente paso es construir la tabla de partición, para esto escoger la opción por defecto, también se puede optar por usar LVM, a este punto se tiene claro cómo crear una tabla de partición con LVM.



Antes de proceder hay que revisar la tabla de partición por defecto; para esto dar enter en “Yes”.



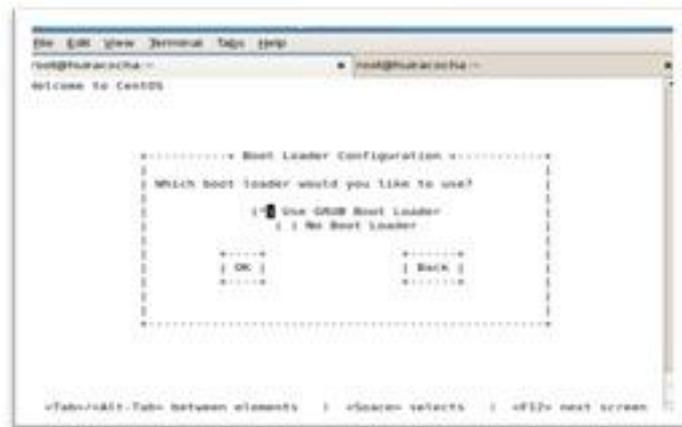
Se puede observar la tabla de partición por defecto; por el momento esto está muy bien; pero se puede optar por algo más sofisticado, se podría optar por colocar algunos directorios en una partición por separado, por ejemplo una partición para home otra para tmp, etc, esto como una recomendación, ya que se puede agregar algunas funcionalidades a nivel de partición, como por ejemplo quotas o también la posibilidad de impedir ejecutar binarios en una partición en especial, como “tmp”.



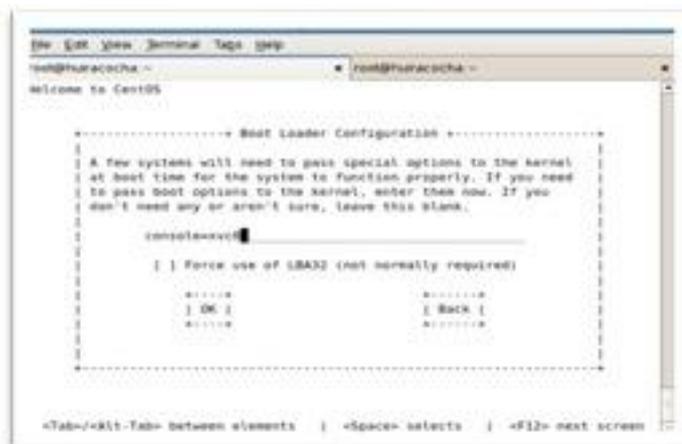
Una vez verificado que la tabla de partición es la misma que la seleccionada o construida procedemos a seleccionar “yes” y dar enter.

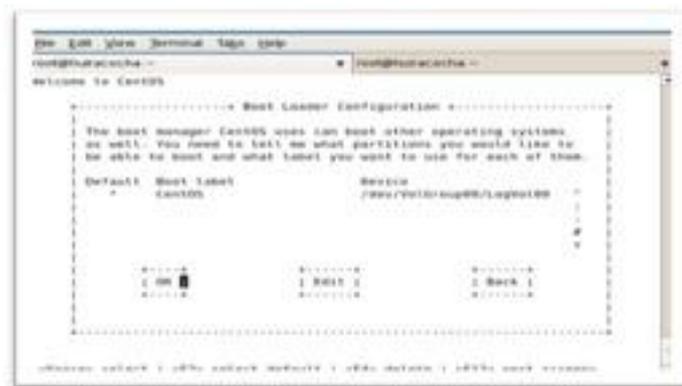


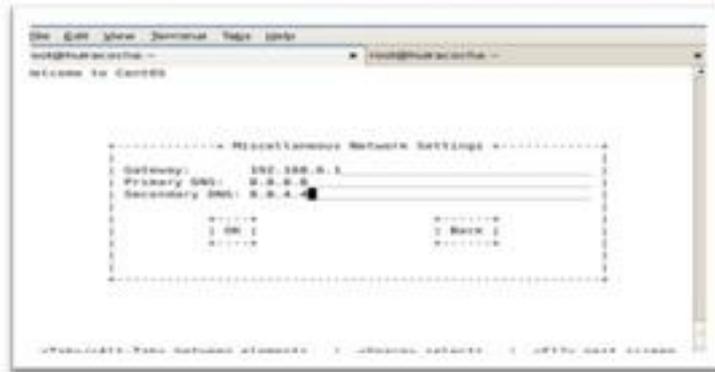
Una vez finalizada la revisión de la tabla de partición, el siguiente paso es seleccionar el gestor de arranque “grub”, una vez seleccionado el gestor, dar enter en OK.



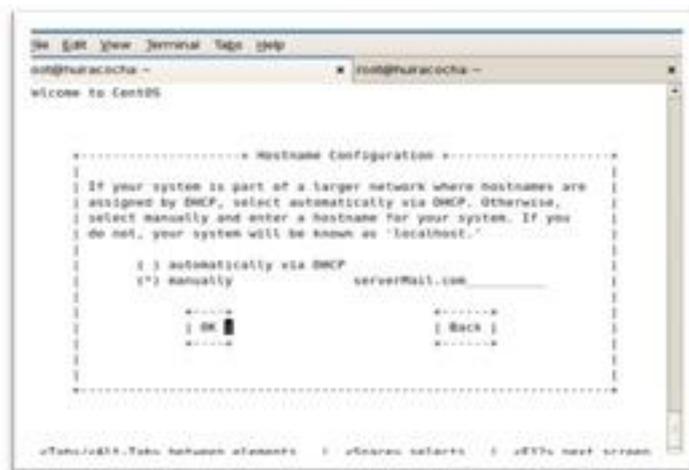
Lo siguiente es seguir con el proceso de instalación por ende se debe seleccionar de acuerdo a lo que indique la imagen.



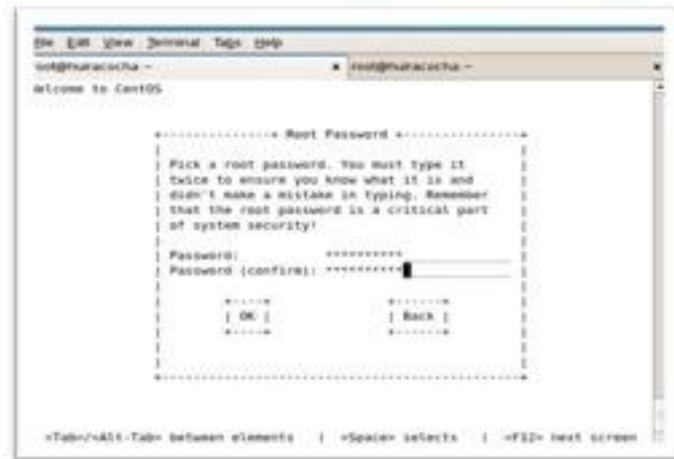




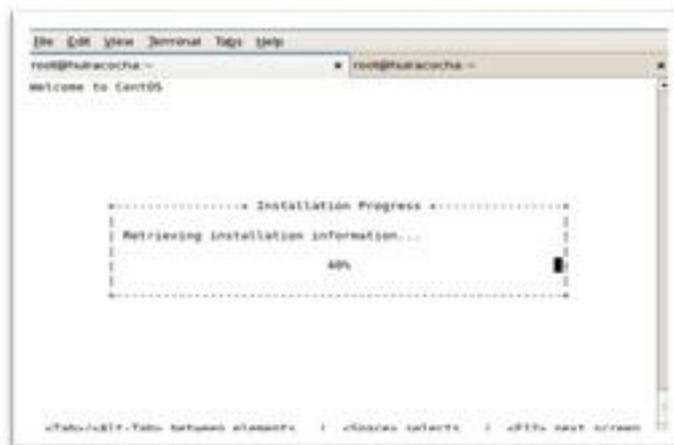
Una vez configurado la red del huésped, escribir el nombre del mismo, como siguiente paso escoger el idioma y tipo de teclado.



Una vez configurado el idioma proceder a configurar la contraseña del usuario root, para ello ingresar dos veces el password del root.



Una vez establecida la contraseña del root esperar hasta que el proceso de instalación recopile información sobre paquetes a instalar.



Una vez finalizada la recopilación de información se procede a seleccionar los paquetes de instalación del huésped, ya que se trata de un servidor seleccionar “server” y quitar “Desktop-Gnome”; se quiere evitar el consumo de recursos innecesarios por ende se opta por instalar el guest sin interfaz gráfica, seleccionar “ok” y dar enter.

```
File Edit View Terminal Help
root@muracoche ~
Welcome to CentOS

----- Package selection -----
|
| The default installation of CentOS includes a set of software
| applicable for general internet usage. What additional tasks
| would you like your system to include support for?
|
|   | Desktop - GNOME  -
|   | Desktop - KDE    #
|   | Server - GAT     |
|   | Server - GAT     v
|
|   | Customize software selection
|
| -----
|   | OK |           | Back |
| -----
|

<Tab>/<Alt-Tab> between elements | <Space> selects | <F12> next screen
```

```
File Edit View Terminal Help
root@muracoche ~
Welcome to CentOS

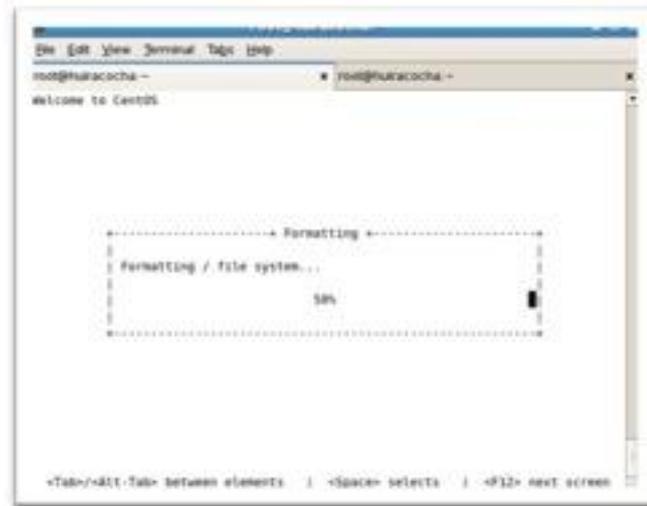
----- Dependency Check -----
|
| Checking dependencies in packages selected for installation...
|
| -----
|           25%
|
| -----

<Tab>/<Alt-Tab> between elements | <Space> selects | <F12> next screen
```

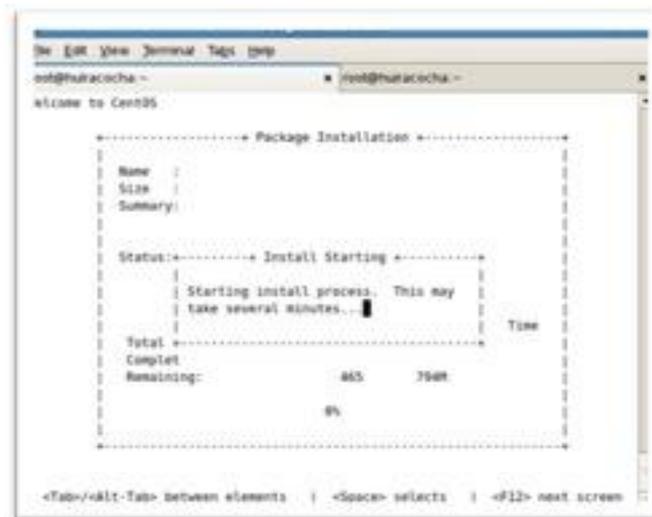
```
File Edit View Terminal Help
root@muracoche ~
Welcome to CentOS

----- Installation to begin -----
|
| A complete log of your installation will
| be in /root/install.log after rebooting
| your system. You may want to keep this
| file for later reference.
|
| -----
|   | OK |           | Back |
| -----
|

<Tab>/<Alt-Tab> between elements | <Space> selects | <F12> next screen
```



En este punto se puede observar que el proceso de instalación del huésped se ha iniciado, simplemente se tiene que esperar hasta que el proceso termine.



Una vez finalizado el proceso de instalación del huésped, se pedirá reiniciar el Sistema Operativo.

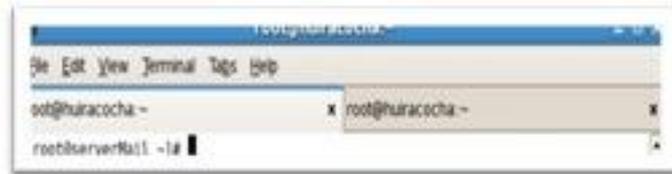
```
root@fuzacoche ~#
root@fuzacoche ~#

sending termination signals...done
sending kill signals...done
stopping tmpfs...
  /dev/mapper/VolGroup00-LogVol01
unmounting filesystems...
  /mnt/sysimage done
  disabling /dev/loop0
  /proc done
  /dev/pts done
  /sys done
  /tmp/ramfs done
  /mnt/iso done
  /mnt/sysimage/lost done
  /mnt/sysimage/ign done
  /mnt/sysimage/proc done
  /mnt/sysimage/selinux done
  /mnt/sysimage/udev done
  /mnt/sysimage done
rebooting system
```

Al reiniciar el Sistema Operativo del huésped, hay que configurar algunas opciones más como son el firewall y el selinux, si nuestro proceso de instalación muestra la siguientes pantalla seleccionar y presionar “Exit”.



Una vez finalizado todo el proceso de instalación ya se tiene instalado el primer “GUEST” con el nombre de “serverMail”.



Para instalar más guest Centos, se vuelve a ejecutar los mismo pasos que en la instalación del primer guest, si desea instalar guest con sistema operativo Windows; por ejemplo un Windows Server, se necesitará soporte de virtualización, para instalarlo en modo full-virtual, al instalar guest en modo virtualización completa, no requiere modificación del kernel y como es bien conocido Windows es un Sistema Operativo privativo por ende no se puede modificar el kernel.

3.7.2 Instalación de un huésped Windows con método fullvirtual

Para ejemplificar el método de full virtualización se usará el Sistema Operativo Windows, dado que Windows entra en la categoría de un sistema operativo sin modificar, es decir que no altera su kernel para fusionarlo con el hypervisor por ende la técnica de paravirtualización no es una opción.

La mejor manera por lo tanto para virtualizar Windows es la virtualización por hardware o también llamado full virtualización.

Para realizar la instalación de Windows en modo full virtualización se verifica que el soporte de virtualización por hardware exista, para ello se escribe el siguiente comando en una consola.

```
# cat /proc/cpuinfo | grep "svm|vmx"
```

Si el comando anterior no produce una salida, es posible que las características de virtualización no están disponibles en el hardware por lo tanto no hay como instalar Windows.

En caso de encontrar las cadenas svm o vmx en el archivo cpuinfo se procede a insertar el DVD de instalación y montar el disco como una imagen ISO, se procede a introducir el siguiente comando en una terminal.

```
# dd if=/dev/cdrom of=windows.iso
```

Como siguiente paso se prepara el espacio de almacenamiento para Windows 7, se puede optar por usar lvm o crear una imagen de disco mediante el comando “dd”, en este caso se creará una imagen de disco de la siguiente forma:

```
# dd if = / dev / zero of = bs = 1024k xenwin7.img seek = 10000 count = 0
```

El comando anterior crea un archivo de imagen de 10 Gb llamado xenwin7.img, a continuación se crea un archivo de configuración o también llamado kickstart que será leído a la hora de instalar Windows, el archivo de configuración contiene los siguientes datos:

```
Os de importación, de nuevo
arco os.uname = () [4]
si re.search ('64 ', arco):
arch_libdir = 'lib64'
otra cosa:
arch_libdir = 'lib'

kernel = "/usr/lib/xen/boot/hvmloader"
constructor = "HVM"
de memoria = 1024

# En caso de ser por lo menos 2 KB por MB de memoria de dominio,
además de unos pocos MB por vcpu.
shadow_memory = 8
name = "xenwin7"
vif = ['type = ioemu, puente xenbr0 =']
acpi = 1
apic = 1
disk = ['file: home/xen/xenwin7.img / hda, w', 'file: /
home/xen/windows7.iso, hdc: cdrom r,']

device_model = '/usr/' + arch_libdir + '/xen/bin/qemu-dm'
```

```
# De arranque en disquete (a), el disco duro (c) o CD-ROM (d)
# por defecto: disco duro, CD-ROM, disquete
boot = "CC"
sdl = 0
vnc = 1
vnconsole = 1
vncpasswd = "

de serie = 'pty "
usbdevice = 'comprimido'
```

En el ejemplo anterior Windows 7 va a ser instalado en disco virtual mediante un medio de instalación que es un archivo de imagen ISO.

Con la configuración del huésped listo, el siguiente paso es iniciar el entorno huésped Xen, para esto introducir el siguiente comando en una consola:

```
xm create xenwin7.cfg

Using config file "./xenwin7.cfg"

Started domain xenwin7
```

Finalmente el proceso de instalación comenzará.

Se ha mostrado el proceso para la instalación de un guest llamado serverMail, se instaló otro guest llamado serverWeb siguiendo el mismo proceso de instalación que serverMail, así mismo se procedió con la instalación de un huésped Windows, como siguiente paso se procede a describir directorios de configuración en xen.

La ruta de configuración del servicio xen se localiza en /etc/xen, dentro de la ruta existen 2 carpetas importantes y archivos de configuración de xen; las carpetas son auto y scripts; en auto se configuran las máquinas que inician en automático cuando el servidor es reiniciando y dentro de la carpeta scripts van ejecutables que necesita xen para configurar algún parámetro de la red virtual o dispositivo que se requiera.

También está el archivo xend-config.sxp, que tiene toda la configuración de la red del bridge, route y nat; también en esta ruta existen archivos de

configuración con los nombres de los servidores virtualizados, como serverMail, serverWeb estos archivos tienen las propiedades de cada servidor virtualizado. Procedemos a observar el contenido de serverMail.

```
name = "serverMail"
```

```
uuid = "2bd420e8-8c10-23a9-cdd2-efbbfed88e68"
```

```
maxmem = 300
```

```
memory = 300
```

```
vcpus = 1
```

```
bootloader = "/usr/bin/pygrub"
```

```
on_poweroff = "destroy"
```

```
on_reboot = "restart"
```

```
on_crash = "restart"
```

```
disk = [ "phy:/dev/dsk/serverMail,xvda,w" ]
```

```
vif = [ "mac=00:16:36:7d:25:d1,bridge=xenbr0,script=vif-bridge" ]
```

Inicio de máquinas virtuales

Para iniciar un Máquina Virtual ejecutar el siguiente comando en una terminal.

```
# xm create serverWeb
```

Al ejecutar el comando `xm create`, crea el dominio e iniciar el GUEST dentro del hypervisor.

Reinicio de máquinas virtuales

Para reiniciar las máquinas virtuales se tiene que usar el siguiente comando.

```
# xm reboot serverWeb
```

Es como si reiniciamos la máquina desde adentro con un `init 6` o `shutdown -r now`.

Destruir de máquinas virtuales

No se recomienda utilizar este comando ya que lo que hace es un apagado de la máquina virtual sin esperar a pagar sus servicios, es como si hubiera un corte de luz o el famoso botonazo.

```
# xm destroy serverWeb
```

Al ejecutar `xm drestroy` hay ocasiones que te crea un respaldo de la máquina virtual en `/var/run/xend` y si tu servidor cuenta con poco espacio va hacer que tu servidores se caigan.

Pausa de máquinas virtuales

Deja la máquina virtual en modo pausa.

```
# xm pause web
```

Con ésta opción no se tiene acceso alguno a la máquina virtual.

Habilitar máquinas virtuales suspendidas

Reinicia la operación de la máquina virtual pausada.

```
# xm unpause web
```

Renombrar máquinas virtuales.

También que puede cambiar el nombre de la una máquina virtual, siempre tener un nombre que relacione con el nombre del servicio que ofrece la máquina virtual.

```
# xm rename serverWeb apache2
```

Monitoreo mv en tiempo real

El siguiente comando sirve para el monitoreo de las VM en tiempo real.

```
# xm top
```

```

root@huiracochoa:/etc/xen
File Edit View Terminal Tabs Help
xentop - 20:55:27 Xen 3.1.2-238.9.1.el5
3 domains: 1 running, 2 blocked, 0 paused, 0 crashed, 0 dying, 0 shutdown
Mem: 1809192k total, 1808872k used, 320k free CPU: 2 @ 2255MHz
S NETS NETTX(k) NETRX(k) VBDS VBD_00 VBD_R0 VBD_WR SSID
Domain-0 -----r 605 14.3 1170596 64.7 no limit n/a
2 4 2 9 0 0 0 0 0
serverMail --b--- 18 37.7 306688 17.0 307200 17.0
1 1 0 2 1 0 2829 860 0
serverWeb --b--- 14 15.0 307836 17.0 307200 17.0
1 1 1 0 1 0 844 0 0
Delay Networks vBds CPUs Repeat header Sort order Quit

```

El comando “top” muestra la cantidad de memoria usada y disponible por el servidor. Cantidad de procesadores que tiene el servidor, también muestra la información de cuantos dominios están configurados; dentro de los dominios configurados se muestra cuantos CPU tiene cada máquina virtual, estado del procesador, memoria total y cuanto ha sido ocupada.

Monitoreo mv en tiempo de inicio

Con este comando nos muestra el tiempo que lleva en línea la MV.

```
# xm uptime
```

```

root@huiracochoa:/etc/xen
File Edit View Terminal Tabs Help
root@huiracochoa xen]# xm uptime
name ID Uptime
Domain-0 0 1:04:41
serverMail 1 0:01:23
serverWeb 2 0:01:01

```

Listar máquinas virtuales

Con el comando “xm list” podremos ver cuántas máquinas virtuales están en ejecución

xm list



Toda la información no las proporciona en columnas, Nombre VM, numero de proceso, memoria, cpu virtuales, estado, tiempo de ejecución. Como pueden ver es información sencilla de entender, solamente la columna de State es la tiene variaciones.

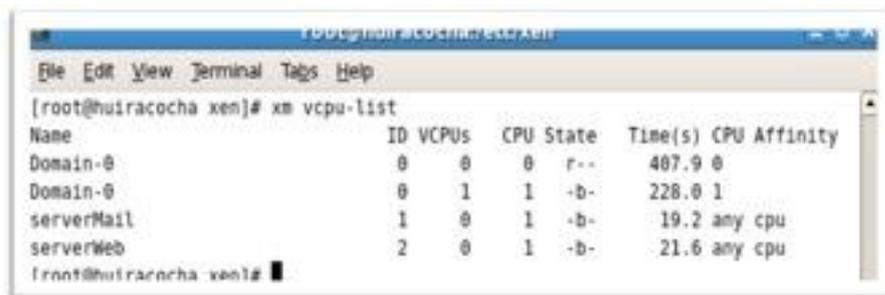
Opción	Descripción
r	El dominio está actualmente en ejecución en el CPU
b	El dominio se encuentra bloqueado, debido que el dominio está en espera de E/S, o no tiene nada que hacer y se va dormir.
p	El dominio se encuentra en pausa, debido que el administrador ejecuto xm pause, en modo pausa el VM seguirá consumiendo los recursos asignados.
s	El administrador ha solicitado un apagado, reinicio o suspendida de la VM.

c	El VM ha tenido un final violento, este estado puede ocurrir si el VM se ha configurado para no volver a arrancar.
d	EL VM está en proceso de morir, pero no se ha pagado por completo. De proceso, memoria, cpu virtuales, estado, tiempo de ejecución. Como

Listar información de la vcpu

Lista información de las vcpu que están ocupando los dominios.

```
# xm vcpu-list
```



```

[root@huiracocha xen]# xm vcpu-list
Name          ID VCPUs  CPU  State  Time(s) CPU Affinity
Domain-0      0  0      0    r--    487.9  0
Domain-0      0  1      1    -b-    228.0  1
serverMail    1  0      1    -b-    19.2   any cpu
serverWeb     2  0      1    -b-    21.6   any cpu

```

También “xm cpu-list”, tiene la opción para desplegar los vcpu de un MV, simplemente se ejecuta el comando de la siguiente manera:

```
# xm vcpu-list serverWeb
```

```
Name      ID  VCPUs  CPU  State  Time(s) CPU Affinity
```

```
serverWeb 24 0 7 -b- 21564.7 any cpu
```

```
serverWeb 24 1 5 -b- 27876.0 any cpu
```

Agregar vcpu a un dominio.

```
# xm vcpu-set serverWeb 2
```

El comando “vcpu-set” agrega otro procesador a la máquina virtual que la requiera. El cambio se verá reflejado hasta reiniciar la máquina virtual y también el servidor XEN.

Agregar vcpu/cpu a un dominio.

```
# xm vcpu-pin serverWeb 1 2
```

El comando “vcpu-pin” ajusta 2 procesadores reales y un procesador virtual. Este cambio se verá reflejado hasta reiniciar el servidor virtual y XEN.

Memoria máxima en vm

```
# xm mem-max serverWeb 1500
```

Ajusta a la máquina virtual como límite máximo de la memoria RAM asignamos 720 MB de RAM, hay que reiniciar para que lo cambios se establezcan.

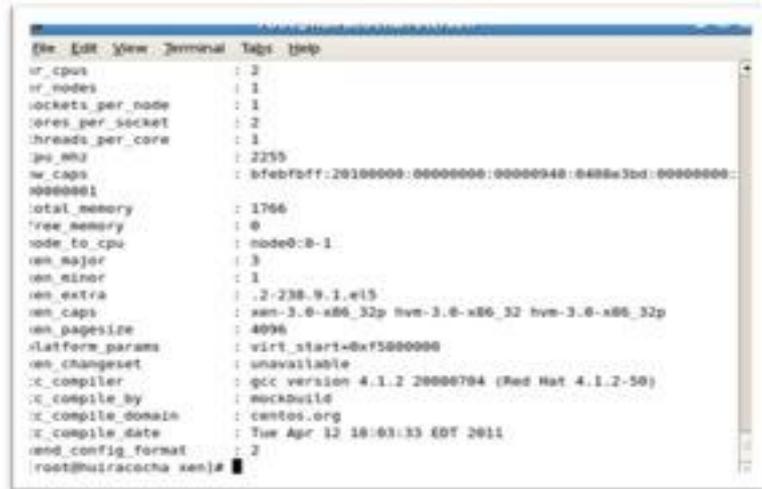
Memoria por defecto ocupada

```
# xm mem-set serverWeb 1024
```

Ajusta a la máquina virtual para que use solamente este espacio asignado en la memoria RAM.

Mostrar información del dominio 0

```
# xm info
```

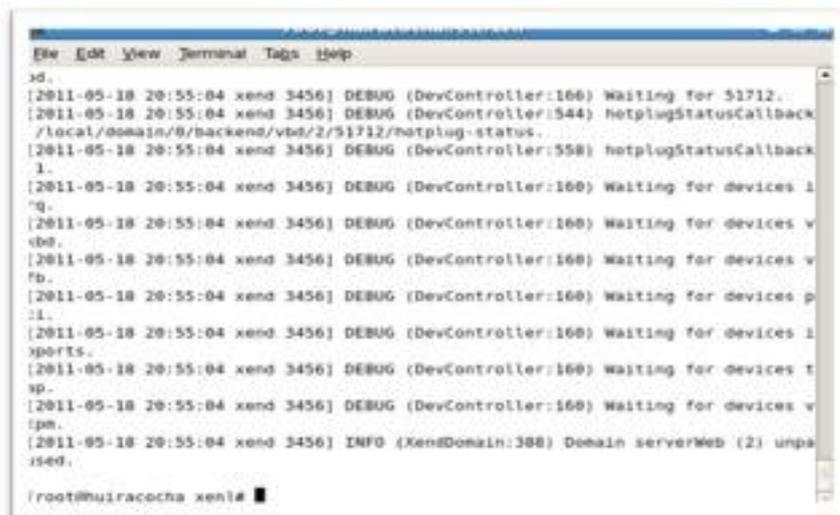


```
File Edit View Terminal Tags Help
nr_cpus          : 2
nr_nodes         : 1
sockets_per_node : 1
cores_per_socket : 2
threads_per_core : 1
ipw_mhz          : 2255
hw_caps          : 8feb7bff:20100000:00000000:00000940:0400x3bd:00000000:
00000001
total_memory     : 1766
free_memory      : 0
node_to_cpu      : node0:0-1
xen_major        : 3
xen_minor        : 1
xen_extra        : .2-238.9.1.e15
xen_caps         : xen-3.0-x86_32p hvm-3.0-x86_32 hvm-3.0-x86_32p
xen_pagesize     : 4096
platform_params : virt_start=0xf5000000
xen_changeset    : unavailable
xc_compiler      : gcc version 4.1.2 20080704 (Red Hat 4.1.2-50)
xc_compile_by    : rockbuild
xc_compile_domain : centos.org
xc_compile_date  : Tue Apr 12 18:03:33 EDT 2011
xen_config_format : 2
root@huiracocho xen#
```

Mostrar logs de xend

Muestra la actividad de los registro de log de Xend.

```
# xm log
```

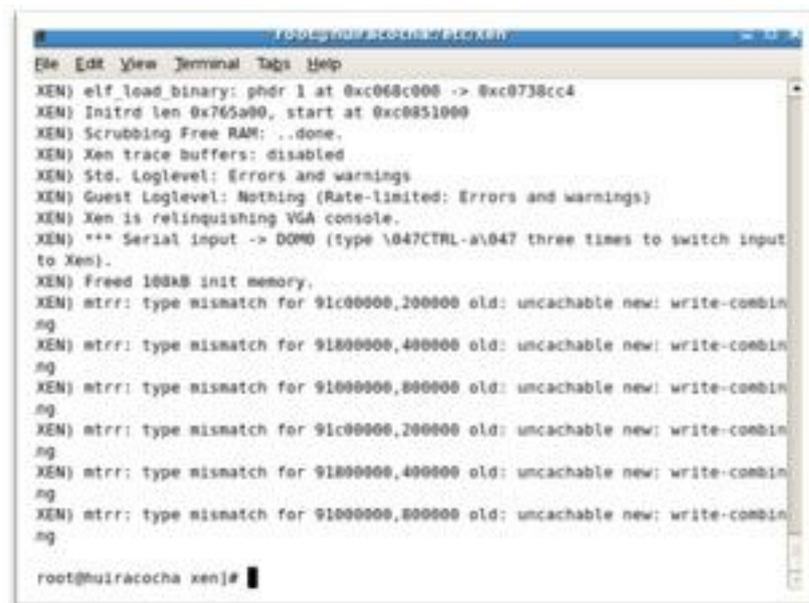


```
File Edit View Terminal Tags Help
xd.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:160) Waiting for 51712.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:544) hotplugStatuscallback
/local/domain/0/backend/vbd/2/51712/hotplug-status.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:558) hotplugStatuscallback
1.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:160) Waiting for devices l
"q.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:160) Waiting for devices v
vbd.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:160) Waiting for devices v
fb.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:160) Waiting for devices p
il.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:160) Waiting for devices l
ports.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:160) Waiting for devices t
sp.
[2011-05-18 20:55:04 xend 3456] DEBUG (DevController:160) Waiting for devices v
ipm.
[2011-05-18 20:55:04 xend 3456] INFO (XendDomain:308) Domain serverWeb (2) unpa
ised.
root@huiracocho xen#
```

Mostrar mensaje del buffer xend

En el buffer contiene la información, advertencias y mensajes de error creados durante el proceso de arranque de XEN o en su caso de que se tenga algún problema con XEN, este es primer lugar que se tiene que verificar.

```
# xm dmesg
```



```
File Edit View Terminal Tags Help
XEN) elf load binary: phdr 1 at 0xc068c000 -> 0xc0738cc4
XEN) Inifrd len 0x765a90, start at 0xc0851000
XEN) Scrubbing Free RAM: ..done.
XEN) Xen trace buffers: disabled
XEN) Std. Loglevel: Errors and warnings
XEN) Guest Loglevel: Nothing (Rate-limited: Errors and warnings)
XEN) Xen is relinquishing VGA console.
XEN) *** Serial input -> DPMO (type \047CTRL-a\047 three times to switch input
to Xen).
XEN) Freed 100kB init memory.
XEN) mtrr: type mismatch for 91c00000,200000 old: uncachable new: write-combin
ng
XEN) mtrr: type mismatch for 91800000,400000 old: uncachable new: write-combin
ng
XEN) mtrr: type mismatch for 91600000,800000 old: uncachable new: write-combin
ng
XEN) mtrr: type mismatch for 91c00000,200000 old: uncachable new: write-combin
ng
XEN) mtrr: type mismatch for 91800000,400000 old: uncachable new: write-combin
ng
XEN) mtrr: type mismatch for 91600000,800000 old: uncachable new: write-combin
ng
root@huiracochoa xen]#
```

3.8 Logs en xen

El componente de virtualización xen contiene el demonio xend y el proceso qemu-dm, dos utilidades que escriben varios archivos de registro en el directorio /var/log/xen:

- xend.log, es el archivo de registro que contiene todos los datos recogidos por el demonio xend, ya sean éstos eventos normales del sistema o acciones iniciadas por el operador. Todas las operaciones sobre la máquina virtual (crear, apagar, destruir, etc.) aparecen aquí.

- `xend-debug.log`, es el archivo de registro que contiene entradas sobre los eventos de error de `xend` y los subsistemas de virtualización (`framebuffer`, `scripts` de `Python`, etc).
- `xen-hotplug-log`, es el archivo de registro que contiene datos sobre los eventos de conexión en caliente. Si un dispositivo o `script` de red no aparece en línea, el evento será registrado en este archivo.
- `qemu-dm.[PID]. Log`, es el archivo de registro creado por el proceso `qemu-dm` para cada huésped virtualizado. Cuando utilice este archivo de registro debe obtener el PID del proceso `qemu-dm` utilizando el comando `ps`. de proceso, memoria, `cpu` virtuales, estado, tiempo de ejecución. Como Examine los argumentos del proceso para aislar el proceso `qemu-dm` en la máquina virtual. Tenga en cuenta que debe reemplazar el símbolo `[PID]` con el PID del proceso `qemu-dm`.

Si encuentra algún error con el administrador de máquinas virtuales, puede revisar los datos generados en el archivo `virt-manager.log` ubicado en el directorio `virt-manager`. Tenga en cuenta que cada vez que inicia el administrador de máquinas virtuales, el contenido del archivo de registro será sobrescrito. Asegúrese de crear una copia de seguridad de `virt-manager.log` antes de reiniciar el administrador de máquinas virtuales después de un error.

3.9 Migración de máquinas virtuales

La migración es el traspaso de un dominio virtual en ejecución o desconectado desde un anfitrión físico a otro. El componente de virtualización `xen` soporta dos clases de migraciones: en vivo y desconectada. La migración desconectada mueve la máquina virtual de un anfitrión a otro deteniendo la máquina, transfiriendo la memoria y reanudando la máquina en el anfitrión de destino. La migración en vivo lleva a cabo la misma tarea pero no afecta directamente al dominio. Cuando se ejecuta una migración en vivo, el dominio continua su actividad usual; por lo cual, esta tarea pasa inadvertida desde la perspectiva del

usuario. El demonio xend y el componente de virtualización xen deben estar en ejecución en los dos anfitriones para poder iniciar una migración en vivo. El anfitrión de destino debe tener recursos suficientes para acomodar el dominio después de la migración. Ambos anfitriones deben tener la misma arquitectura y las mismas extensiones de virtualización (como i386-VT, x86-64-VT, x86-64-SVM, etc.). Así mismo, los dos anfitriones deben ser parte de la misma subred.

Ya que estos dominios tienen una carga de archivos grande, este proceso puede tomar bastante tiempo. Si se migra un dominio con conexiones de red abiertas, éstas serán preservadas al anfitrión de destino; las conexiones SSH seguirán funcionando. Las reglas iptables predeterminadas del componente de virtualización de Red Hat no permitirán conexiones de migración entrantes. Para permitir las se deberá crear una regla iptables explícita.

Puede usar el comando migrate de xm para ejecutar una migración desconectada:

```
# xm migrate domain-id [destinación domain]
```

Puede utilizar el comando migrate de xm con la opción “-l” para ejecutar una migración en vivo:

```
# xm migrate domain-id -l [destinación domain]
```

Ejemplo migración en vivo:

Una maravilla que nos permite Xen es el hacer migración en vivo entre dos máquinas. Una migración en vivo, es la capacidad de mover una máquina virtual entre dos máquinas físicas sin que esto se note. ¿Para qué sirve esto? Imaginaremos que tenemos un servidor corriendo en un entorno virtual y queremos cambiar su disco para aumentar el tamaño entre otras cosas, sin tener que sacrificar el servicio, cambiamos la máquina virtual con el live

migration, sin dejar en ningún momento el servicio parado y hacemos lo que tengamos que hacer.

Para que funcione el esquema de Migración en Vivo se necesitan 2 servidores xen, el primer servidor contendrá las máquinas a migrar, lo llamaremos servidorA un segundo servidor, servidorB contendrá las imágenes y una esquema con servicio xen.

El esquema es el siguiente:

ServidorA(192.168.6.3):

entorno xen

mount 192.168.6.4:/xen en /dev/dsk/

serverMail corriendo.

ServerWeb corriendo.

ServidorB(192.168.6.4):

entorno xen

mount 192.168.6.4:/xen en /dev/dsk

serverNFS corriendo.

Compartimos el directorio /xen

(migración en vivo) serverMail.

Se procede a configurar el servicio nfs en el servidor B para ello introducimos los siguientes comandos en una terminal.

```
# yum -y install portmap nfs-utils

# service portmap start

# service nfslock start

# service nfs start

# mkdir /xen

# vi /etc/exports

/xen 192.168.6.0/24(rw,async,no_root_squash)

# exportfs -ra
```

En el archivo de configuración /etc/exports, se especifica la red en la que se encuentra el servidor, es decir que tanto servidor A como servidor B se encuentran en el mismo segmento de red; se procede a enumerar algunas opciones al momento de configurar nfs:

ro: Solo lectura.

rw: Lectura Escritura.

async: Escritura asíncrona.

sync: Escritura síncrona.

no_root_squash: Permitir escritura de archivos y asignaros a root.

root_squash: No Permitir escritura como root.

Una vez configurado el servidor nfs se procede a montar el directorio compartido “/xen” en el directorio local “/dev/dsk”.

```
mount 192.168.6.4:/xen /dev/dsk
```

Una vez montado el directorio compartido “xen”, proceder a configurar el servidor xen para que acepte peticiones de relocalización de máquinas virtuales desde la ip 192.168.6.3, para esto se realiza lo siguiente desde una terminal:

```
#vi /etc/xen/xend-config.sxp
```

```
(xend-relocation-port                8002)
(xend-relocation-address             '192.168.6.4')
(xend-relocation-hosts-allow '192.168.6.3')
```

Una vez finalizada la configuración del servidor B, proceder a configurar el servidor A.

Como primer paso, instalar el servicio nfs el servidor B, seguimos los siguientes pasos:

```
# yum -y install portmap nfs-utils
```

```
# service portmap start
```

```
# service nfs-utils
```

Se puede observar que no es necesario levantar el servicio nfslock ya que se trata del cliente.

Ahora se procede a montar el directorio compartido xen en /dev/dsk.

```
# mount 192.168.6.4:/xen /dev/dsk
```

Un problema que se puede presentar es al reiniciar el servidor físico se debe arrancar los servicios y montar los directorios, este problema se puede solucionar con chkconfig y con el fstab.

Procedemos a configurar el servidor A, para que acepte peticiones de envío, de máquinas virtuales hacia la ip 192.168.6.4, para esto realizar los siguientes pasos:

```
#vi /etc/xen/xend-config.sxp
```

```
(xend-relocation-port                8002)
(xend-relocation-address              '192.168.6.4')
(xend-relocation-hosts-allow '192.168.6.3')
```

Una vez configurado el servidor “A” y el servidor “B” proceder con la migración en vivo, se enviará el huésped que se encuentra en servidor A hacia el servidor B, para realizar esto ejecutar el siguiente comando desde una terminal.

```
# xm migrate serverMail -l 192.168.6.4
```

Finalizada la migración en vivo comprobar que el huésped efectivamente se encuentra migrado en el servidor “B” introduciendo el siguiente comando.

```
# ls /etc/xen/serverMail
```

3.10 Inicio automático de los domU

Puede arrancar dominios huéspedes a través de la aplicación xm. También se puede utilizar virsh y el administrador de máquinas virtuales para arrancar sistemas huéspedes. Este ejemplo utiliza el subcomando create de xm:

```
# xm create -c serverWeb
```

El nombre “serverWeb” del comando introducido anteriormente, es el archivo de configuración para el dominio que se está iniciando. La opción -c se conecta a la consola actual después del proceso de arranque, este proceso se lo tiene que realizar cada vez que se reinicie el hypervisor.

Se puede iniciar o detener un dominio en cualquier momento. Domain0 espera que todos los dominios en ejecución se detengan antes de reiniciar. Se debe ubicar los archivos de configuración del dominio que desea apagar en el directorio /etc/xen/. Todos los dominios que serán iniciados durante el proceso de arranque deben tener un enlace simbólico en /etc/xen/auto.

```
# chkconfig xendomains on
```

El comando “chkconfig” xendomains on, no iniciará los demonios automáticamente. Este comando indica que los demonios deben ser iniciados en arranques subsecuentes.

```
# chkconfig xendomains off
```

Termina todos los dominios de virtualización de Red Hat en ejecución. El comando chkconfig xendomains off indica que los dominios no deben ser iniciados en los próximos arranques.

3.11 Tarjetas de red en XEN.

XEN crea por defecto interfaces Ethernet virtuales interconectadas para que dom0 las utilice. Si tecleamos el comando ifconfig en el servidor xen, muestra las siguientes interfaces de red.

1. Interfaz peth0.
2. Interfaz xenbr0.
3. Interfaz virbr0
4. Interfaz vif#.#

Se procede a describir las interfaces de red mostradas con el comando “ifconfig”.

peth0

La interfaz peth0 “Physical Ethernet” deberíamos conocerla como interfaz física, lo que hace es renombrar la interfaz eth0 por peth0 para que sea ocupada por XEN en modo bridge. La interfaz eth0 es desactiva en el Dom0 y es renombrada a peth0.

```
peth0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF
```

```
UP BROADCAST NOARP MTU:1500 Metric:1
```

```
RX packets:49643901 errors:0 dropped:0 overruns:0 frame:0
```

```
TX packets:48752155 errors:1293908 dropped:0 overruns:0  
carrier:1293908
```

collisions:2057033 txqueuelen:1000

RX bytes:26327146789 (24.5 GiB) TX bytes:31709052034
(29.5 GiB)

Interrupt:16 Memory:f8000000-f8012100

xenbr0

La interfaz xenbr0, también es conocido como switch virtual, el cual permite conectar a las máquinas virtuales en una red local o DMZ, pero que están controlados por un administrador de red que les asigna ip estáticas o dinámicas.

xenbr0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF

UP BROADCAST RUNNING NOARP MTU:1500 Metric:1

RX packets:316559 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:23948805 (22.8 MiB) TX bytes:0 (0.0 b)

virbr0

La interfaz virbr0 “virtual bridge”, es el puente de la red virtual creada por xen, todas las máquinas virtuales alojados por esta interfaz será parte de una red virtual (route, nat) o DMZ virtual.

virbr0 Link encap:Ethernet HWaddr 00:00:00:00:00:00

inet addr:192.168.122.1 Bcast:192.168.122.255
Mask:255.255.255.0

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

Vif0.#

La interfaz vif0.#, es el puente del puerto que es utilizado por el tráfico al del Dom0. Los dispositivos de red virtual son las conexiones hacia las tarjeta de red de las máquinas virtuales, El símbolo # toma su valor dependiendo del ID de la máquina virtual al iniciar.

vif0.0 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF

UP BROADCAST RUNNING NOARP MTU:1500 Metric:1

RX packets:1417928 errors:0 dropped:0 overruns:0 frame:0

TX packets:2240859 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:210227871 (200.4 MiB) TX bytes:1181653865 (1.1 GiB)

vif0.21 Link encap:Ethernet HWaddr FE:FF:FF:FF:FF:FF

UP BROADCAST NOARP MTU:1500 Metric:1

RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:32

RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

3.12 Network xen

Xen tiene la propiedad de configurar redes virtuales dentro de los propios d6minos. Existen varias configuraciones de red en XEN:

1. Puente (bridge).
2. Encaminador (router).
3. NAT.
4. otras.

6sta configuraci6n se lleva a cabo con las interfaces de red virtuales, pero tambi6n es necesario configurar el archivo `/etc/xen/xend-config.sxp`, para poder hacer el cambio a la configuraci6n de la red.

Puente (bridge)

La configuraci6n por defecto de XEN crea puentes de red dentro de dom0 para permitir que todos los dominios aparezcan en la red como hosts independientes. Este m6todo lo que hace es tomar IP dentro de la red donde

se conecta, ya sea IP estática o dinámica. Este es el método más simple y fácil de configurar la red para las máquinas virtuales, simplemente permite que sus máquinas virtuales utilicen una tarjeta Ethernet virtual para unirse a una red existente.

Para poder tener esta configuración solamente tenemos que des comentar las siguientes líneas del archivo de configuración de XEN /etc/xen/xend-config.sxp.

```
(network-script network-bridge)
```

```
(vif-script vif-bridge)
```

verificar nuestra MV tenga la siguiente línea.

```
vif = [ "mac=00:16:3e:56:16:d6,bridge=Xenbr0" ]
```

Y finalmente reiniciar XEN para que tome los cambios.

bridge varias ethernet

Este es un método por el cual podemos tener máquinas virtuales, en varias interfaz xenbr# y peth#, lo cual puede mejora el rendimiento del tráfico de las DomU. Para esto haremos lo siguiente, crearemos el siguiente script y lo guardados en la ruta /etc/xen/scripts/

```
# vim puentes.sh
```

```
#!/bin/sh
```

```
dir=$(dirname "$0")
```

```
"$dir/network-bridge" "$@" vifnum=0 netdev=eth0  
bridge=xenbr0
```

```
"$dir/network-bridge" "$@" vifnum=1 netdev=eth1  
bridge=xenbr1
```

Con esto hemos creado dos bridges dentro de XEN. Ahora solo tenemos que modificar el archivo de configuración de XEN.

```
# vim /etc/xend/xend-config.sxp
```

Modificar el siguiente parámetro:

```
 #(network-script network-bridge)
```

Y agregar esta línea debajo de la comentada:

```
(network-script puentes.sh)
```

Listo ya tenemos configurado xen con dos puentes, solamente queda reiniciar el servicio xen para que tome los cambios.

```
/etc/init/xend restart
```

Encaminador (router)

Esta configuración se aplica cuando:

- Cuando la máquinas están en una diferente LAN.
- Cuando el tráfico se red dirige hacia el exterior por la red 192.168.1.0/24
- Son máquinas visibles desde 192.168.1.0/24

Para poder hacer esta configuración es necesario modificar el archivo de configuración de XEN. Buscar las siguientes líneas y des comentarlas.

```
(network-script network-route)
```

```
(vif-script vif-route)
```

Comentar las siguientes:

```
 #(network-script network-bridge)
```

```
 #(network-script network-nat)
```

```
 #(vif-script vif-bridge)
```

```
 #(vif-script vif-nat)
```

Ya con esto se tendrá habilitas el nodo route en xen, pero todavía falta configurar las tarjetas virtuales de las máquinas virtuales, ahora se tiene que modificar archivo de una máquina virtual, para el ejemplo se trate de un servidor web. Como pueden ver en esta línea del archivo de la MV web tiene habilitado que usa la configuración de red en bridge.

```
vif = [ "mac=00:16:3e:56:16:d6,bridge=Xenbr0" ]
```

Tenemos que modificarla o mejor comentarla y crear otra línea debajo de esta con los siguientes datos

```
vif = [ "ip=10.0.0.2" ]
```

Ahora tenemos que habilitar el route siguiente forma:

```
echo 1 > /proc/sys/net/ipv4/conf/all/proxy_arp
```

```
iptables -t nat -A POSTROUTING -s 10.0.0.0 -j MASQUERADE
```

```
route add -net 10.0.0.0 netmask 255.255.255.0 gw 192.168.1.5
```

Ya con esto ya tendrás configurado la red configurado en route solo reinicia XEN.

NAT

Porque usar nat?:

- DomU están en una LAN privada.
- DomU hacen NAT con Dom0 para poder llegar a la otra LAN y parece como si el tráfico procede de Dom0.
- DomU están ocultos y protegidos de 192.168.1.0/24

Para poder habilitar esta configuración tenemos que editar el archivo de configuración de XEN /etc/xen/xend-config.sxp.

Buscar las siguientes líneas y des comentarlas.

```
(network-script network-nat)
```

```
(vif-script vif-nat)
```

Comentar las siguientes:

```
 #(network-script network-bridge)
```

```
 #(network-script network-route)
```

```
 #(vif-script vif-bridge)
```

```
 #(vif-script vif-route)
```

Hay que modificar la configuración de la MV.

```
 vif = [ "mac=00:16:3e:56:16:d6,bridge=Xenbr0" ]
```

Tenemos que modificarla o mejor comentarla y crear otra línea debajo de esta con los siguientes datos

```
 vif = [ "ip=10.0.0.2" ]
```

Con esta modificación ya se tiene la MV trabajando, solo hay que reiniciar XEN.

3.13 Herramienta brctl

Brctl, es una herramienta de red que inspecciona y modifica la configuración de los puentes Ethernet en el Kernel de virtualización de Linux. Debe tener acceso de root antes de ejecutar estos comandos de ejemplo: Este comando nos muestra la información de los puentes y las tarjetas que están conectada de proceso, memoria, cpu virtuales, estado, tiempo de ejecución.

```
 # brctl show
```

```
 bridge name    bridge id STP enabled    interfaces
```

```
 virbr0 8000.000000000000 yes
```

```
 xenbr0 8000.46c9342a9baa no          peth0
```

```
 tap2
```

vif0.28

tap3

vif0.24

tap0

vif0.21

tap1

La opción “macs” del comando brctl, nos muestra la información de las MAC que tiene el puente, también indica que MAC's ya están ocupadas.

```
brctl show macs xenbr0
```

port	no. mac addr	is local?	ageing timer
2	00:01:69:00:3c:dd	no	0.00
2	00:01:69:00:3c:f6	no	0.04
2	00:03:ba:5c:9e:a1	no	199.02
2	00:0c:29:22:94:2f	no	224.37
2	00:14:22:0f:50:75	no	6.54
2	00:14:22:10:16:cd	no	272.70
2	00:16:35:5c:e4:6a	no	45.61

7	00:16:3e:27:f6:55	no	2.16
5	00:16:3e:30:92:bc	no	0.04
3	00:16:3e:56:16:d6	no	0.00
2	00:19:30:9c:ff:9c	no	0.85

CAPITULO 4

ADMINISTRACIÓN DE MÁQUINAS VIRTUALES CON VIRSH

4.1 Introducción

Virsh es una herramienta de línea de comando para administrar a los huéspedes y al hypervisor.

La herramienta virsh se crea en la API de administración libvirt y funciona como una alternativa para el comando xm y el Administrador de huésped gráfico (virt-manager). Virsh puede ser utilizado en modo de sólo lectura por usuarios sin privilegios. Se puede utilizar virsh para ejecutar scripts para las máquinas de huésped, [RHI2010].

La siguiente tabla es una rápida referencia para todas las opciones de línea de comandos de virsh.

Comando	Descripción
setmem	Establece la memoria asignada para un huésped.
setmaxmem	Establece el límite máximo de memoria para el hypervisor.
setvcpus	Cambia el número de CPU virtuales asignadas a un huésped.

vcpuinfo	Muestra información de CPU virtual sobre un huésped.
vcupin	Controla la afinidad de CPU virtual de un huésped.
dombkstat	Muestra las estadísticas de dispositivo de bloque para un huésped en ejecución.
domifstat	Muestra estadísticas de interfaz de red para un huésped en ejecución.
Attach-device	Conecta un dispositivo a un huésped, mediante la definición de un dispositivo en un archivo XML.
attach-disk	Conecta un nuevo dispositivo de disco para un huésped.
attach-interface	Conecta una nueva interfaz de red para un huésped.
detach-device	Desconecta un dispositivo de un huésped, adquiere la misma clase de descripciones del comando attach-device.
detach-interface	Desconecta una interfaz de red de un huésped.
version	Muestra la versión de virsh

nodeinfo	Entrega información acerca del hypervisor

4.2 Conexión al hypervisor

Puede utilizar virsh para iniciar una sesión del hypervisor:

```
virsh connect <name>
```

Donde “<name>” es el nombre de la máquina del hypervisor. Para iniciar una conexión de sólo-lectura, añade al comando anterior la opción readonly.

4.3 Creación de una máquina virtual

Libvirt se puede usar desde la Shell con el comando virsh. Este comando se puede utilizar para administrar los dominios de una manera comparable a xm, Puede crear una nueva sesión de una máquina virtual desde una definición de la máquina en XML.

Los huéspedes pueden ser creados desde los archivos de configuración XML. Se pueden copiar los XML existentes de los huéspedes creados anteriormente, o utilizar la opción dumpxml. Para crear un huésped desde un archivo XML con virsh introducimos el siguiente comando:

```
virsh create <xmlfile>
```

En donde <xmlfile> es el path del archivo xml que contiene la configuración del guest.

4.4 Configuración de un volcado xml

Puede utilizar virsh para ejecutar un volcado de datos para una máquina virtual existente, es decir a partir de un guest existente, volcamos su configuración a un archivo xml, la sintaxis de un volcado de memoria es la siguiente:

```
# virsh dumpxml [domain-id | domain-name | domain-uuid]
```

El comando virsh ofrece el archivo de configuración XML del huésped de manera estándar (stdout). Puede almacenar los datos obtenidos realizando un pipe de la salida hacia un archivo. A continuación ofrecemos un ejemplo de realizar esto hacia un archivo denominado guest.xml:

```
# virsh dumpxml GuestID > guest.xml
```

Este archivo guest.xml puede volver a crear el huésped. Puede editar este archivo de configuración XML para configurar dispositivos adicionales o para utilizar huéspedes adicionales.

Un ejemplo de salida de virsh dumpxml:

```
# virsh dumpxml r5b2-mySQL01
```

```
<domain type='kvm' id='13'>
```

```
  <name>r5b2-mySQL01</name>
```

```
  <uuid>4a4c59a7ee3fc78196e4288f2862f011</uuid>
```

```
  <bootloader>/usr/bin/pygrub</bootloader>
```

```
  <os>
```

<type>linux</type>

<kernel>/var/lib/libvirt/vmlinuz.2dgnU_</kernel>

<initrd>/var/lib/libvirt/initrd.UQafMw</initrd>

<cmdline>ro root=/dev/VolGroup00/LogVol100 rhgb
quiet</cmdline>

</os>

<memory>512000</memory>

<vcpu>1</vcpu>

<on_poweroff>destroy</on_poweroff>

<on_reboot>restart</on_reboot>

<on_crash>restart</on_crash>

<devices>

<interface type='bridge'>

<source bridge='br0'>

<mac address='00:16:3e:49:1d:11'>

<script path='bridge'>

</interface>

```
<graphics type='vnc' port='5900'/>
```

```
<console tty='/dev/pts/4'/>
```

```
</devices>
```

```
</domain>
```

4.5 Suspensión de una máquina virtual

Puede utilizar `virsh` para suspender un dominio, escribiendo el siguiente comando en una terminal:

```
virsh suspend [domain-id | domain-name | domain-uuid]
```

Cuando un dominio se encuentra en un estado de suspensión, todavía consume memoria RAM del sistema. También no habrá ningún disco o red de E / S mientras permanezca en este estado. Esta operación es inmediata y la máquina virtual debe reiniciarse con la opción de reanudar.

4.6 Reanudación de una máquina virtual

Puede utilizar `virsh` para reanudar una máquina virtual suspendida, introduciendo el siguiente comando en una terminal:

```
virsh resume [domain-id | domain-name | domain-uuid]
```

Esta operación es inmediata y los parámetros de la máquina virtual son preservados en un ciclo entre suspendidos y reanudados.

4.7 Guardar una máquina virtual

Puede utilizar virsh para guardar el estado actual de una máquina virtual, escribiendo el siguiente comando en una terminal:

```
virsh save [domain-name][domain-id | domain-uuid][filename]
```

4.8 Apagado de una máquina virtual

Puede utilizar virsh para apagar una máquina virtual, con la siguiente sintaxis:

```
virsh shutdown [domain-id | domain-name | domain-uuid]
```

Se puede controlar el comportamiento de la máquina virtual que será apagada si se modifica el parámetro `on_shutdown` del archivo `xmdomain.cfg`.

4.9 Convertir un nombre de dominio a un ID de dominio

Puede utilizar virsh para convertir un nombre de dominio en un UUID o ID de dominio con la siguiente instrucción:

```
virsh domid [domain-name | domain-uuid]
```

4.10 Ver información de una máquina virtual

Puede utilizar virsh para ver información de una máquina virtual identificada con su ID de dominio, nombre de dominio o UUID, de la siguiente manera:

```
virsh dominfo [domain-id | domain-name | domain-uuid]
```

CAPITULO V

ADMINISTRACIÓN DE MÁQUINAS VIRTUALES CON VIRT-MANAGER

5.1 Introducción

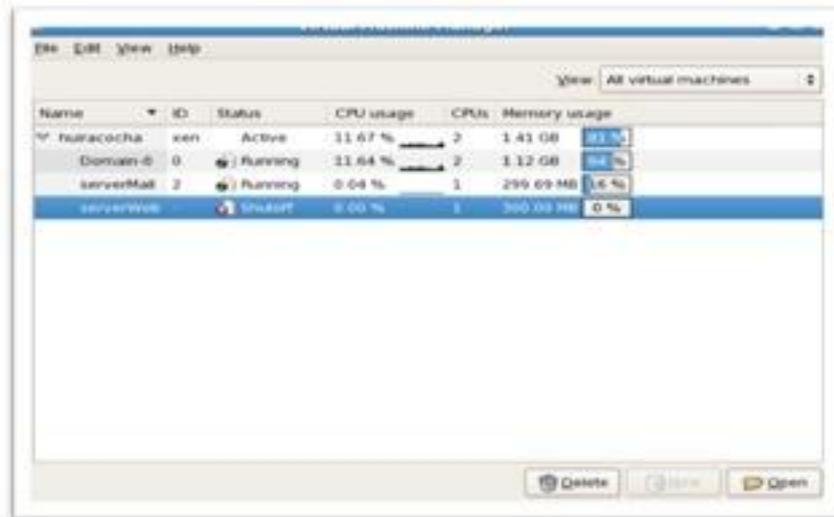
Este capítulo describe las ventanas, cajas de diálogos y otros controladores de la interfaz de usuario del administrador de máquinas virtuales (VMM por sus siglas en inglés) del componente de virtualización xen.

El componente de virtualización xen es una colección de componentes de software que funcionan juntos para hospedar y administrar máquinas virtuales. El administrador de máquinas virtuales (VMM) ofrece una vista gráfica de las máquinas virtuales en su sistema. Se puede utilizar VMM para definir tanto máquinas virtuales paravirtualizadas como completamente virtualizadas. Con el administrador de máquinas virtuales se pueden ejecutar tareas de administración como asignar memoria, asignar CPUs virtuales, monitorizar el rendimiento operacional, guardar, restaurar, detener, resumir y apagar sistemas virtuales. También permite acceder a la consola de texto y a la gráfica. La tecnología de virtualización xen abstrae los recursos de memoria y CPU desde la configuración de red y el hardware subyacente. Esto permite reunir y asignar dinámicamente los recursos de procesamiento a las solicitudes de aplicaciones y servicios. La virtualización a nivel de chips permite que los sistemas operativos que corren en hardware Intel VT y AMD Pacifica puedan ser ejecutados en hipervisores, [RHI2010].

5.2 Ventana para administrador Máquinas Virtuales

Para visualizar el administrador gráfico mediante virt-manager procedemos a ejecutar el siguiente comando:

```
# virt-manager
```

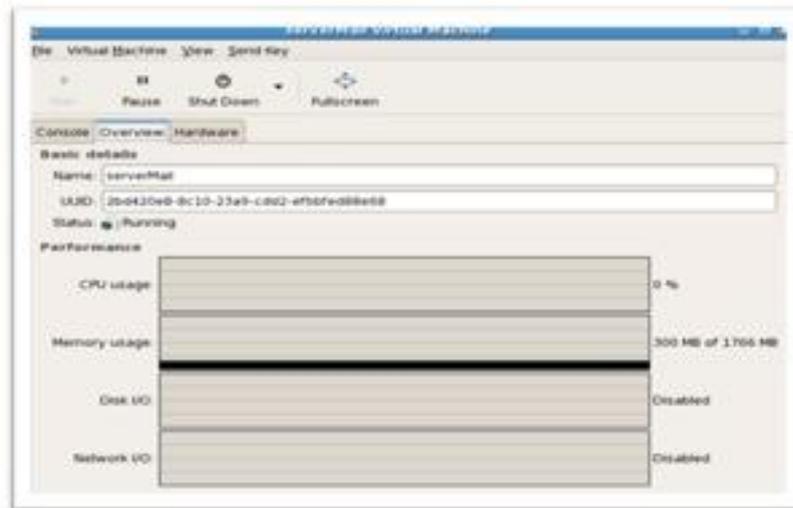


La ventana principal muestra todas las máquinas virtuales y los recursos asignados a éstas, incluyendo el domain0.

También se puede mostrar información detallada por cada máquina virtual que se tiene instalado en el hypervisor xen, para ello dar doble clic sobre la máquina virtual deseada y después a la pestaña Detalles.

5.3 Ventana de detalles de la Máquina Virtual

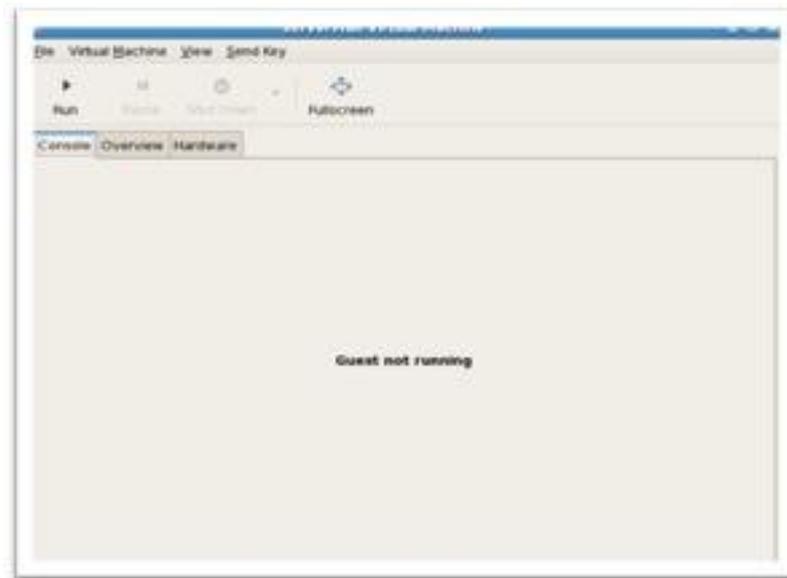
La ventana de detalles muestra un gráfico en tiempo real de los recursos usados por la máquina virtual, se puede observar el uso del cpu y la memoria también se muestra el estado de la máquina virtuales en este caso la máquina virtual se encuentra en el estado (Running).



5.4 Consola gráfica de la Máquina Virtual

La ventana de consola gráfica permite configurar un huésped ya sea paravirtual o full virtual ambas tecnologías están soportadas en la consola gráfica, para esto en cada huésped se debe activar vnc para permitir a la consola administrar los recursos del huésped.

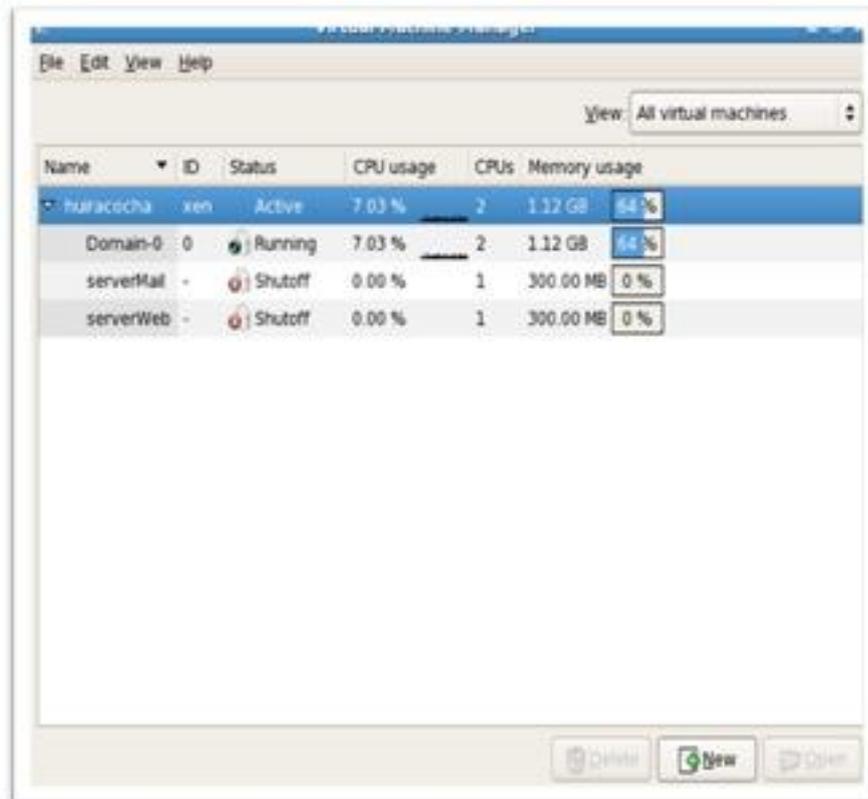
La consola de administración gráfica es compatible con las 'sticky key' se puede enviar contol+alt+supr sin ningún problema, así como muchas otras combinaciones de teclas.



5.5 Crear una nueva Máquina Virtual

Para crear una máquina virtual desde virt-manager realizar los siguientes pasos.

En primera instancia seleccionar el dominio xen, y luego damos click en el botón new.



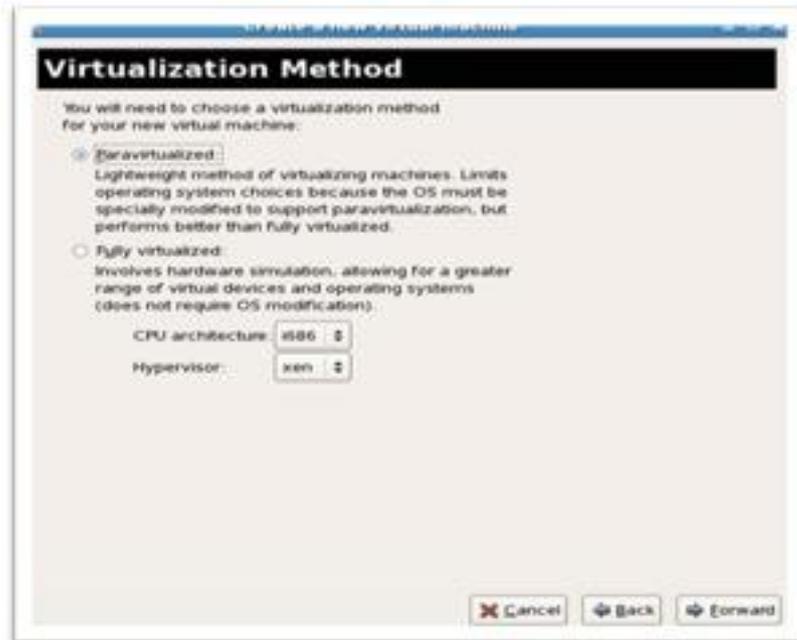
Al comenzar con la instalación de un huésped en entorno gráfico, se visualiza un pantalla con los posibles pasos a seguir, en esta ventana presionamos el botón “Forward”



El siguiente paso es ingresar el nombre del huésped para el ejemplo escribir serverJboss, una vez escrito el nombre dar click en el botón “forward”.



Como siguiente paso escoger el método de virtualización ya se tienen claras las condiciones para cada método, en este caso seleccionar “Paravirtualized”, damos click en el botón siguiente.



Una vez seleccionado el método de virtualización procedemos a escoger la opción “Network install tree” y dar click en el botón siguiente.



Como siguiente paso escribir la URL del repositorio que contiene los archivos necesarios para la instalación del huésped, opcionalmente podemos ingresar un archivo con las configuración básica del huésped este archivo se lo suele llamar kickstart, también se puede enviar algunos parámetros al kernel; procedemos a dar click en el botón “forward”



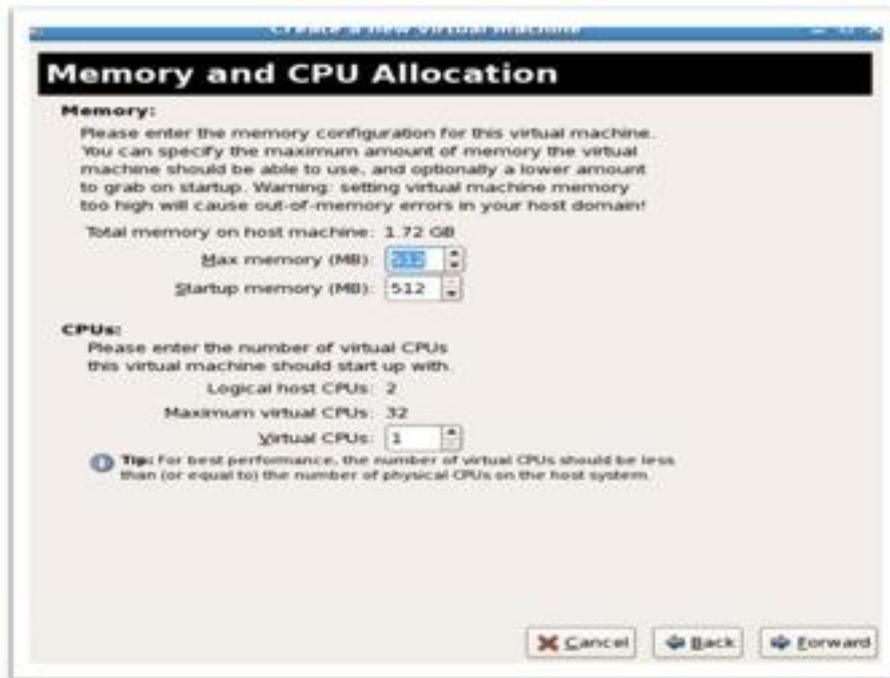
Como siguiente punto se pide ingresar el disco virtual en donde se instalará el huésped, seleccionar un “lv” previamente preparado para la instalación del huésped, proceder a dar click en el botón “forward”.



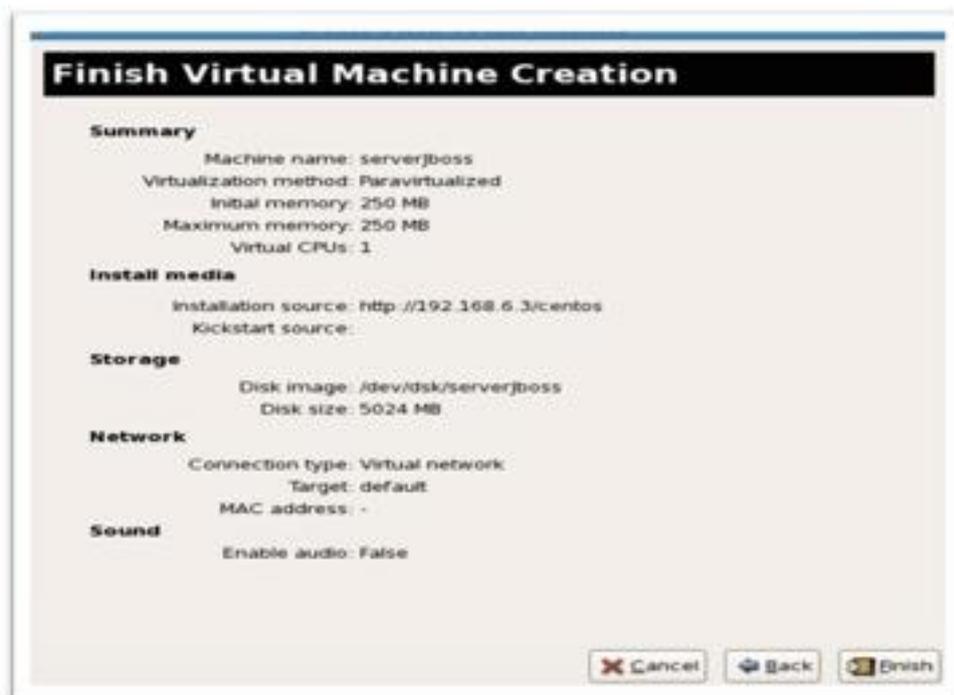
Proceder a configurar el tipo de interfaz de red, seleccionar virtual network, y dar click en el botón “siguiente”.



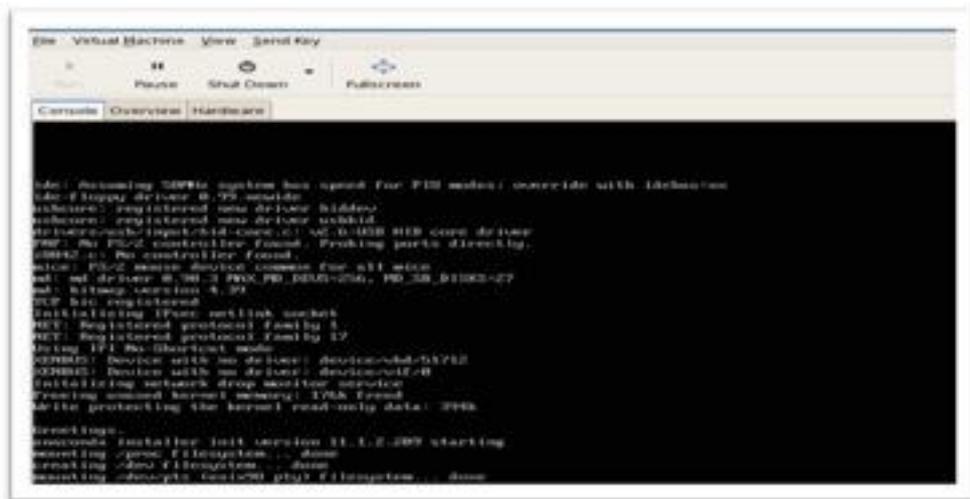
Como siguiente paso configurar la memoria y cpus que serán asignados al huésped, proceder a dar click en el botón siguiente.



Una vez configurado los parámetros generales de un huésped dar click en el botón “Finish”.



El siguiente paso es instalar el sistema operativo normalmente.



Una vez finalizada la instalación del huésped en modo gráfico, proceder a configurar o cambiar algunos parámetros como procesador, memoria, discos, etc. La consola de administración gráfica tiene la característica de brindar herramientas que faciliten el trabajo observar la pestaña de hardware esta nos permite asignar recursos a los huéspedes.



CAPITULO VI

DETECCIÓN Y CORRECCIÓN DE ERRORES.

6.1 Introducción

Este capítulo aborda problemas potenciales que puede experimentar durante la instalación, administración y operaciones diarias de su sistema de virtualización xen. Esta sección describe los mensajes de error, la ubicación de los archivos de registro, las herramientas del sistema y los enfoques generales para analizar los datos obtenidos.

Sinopsis y ubicación de los archivos de registro

El directorio de configuración principal del componente de virtualización xen es `/etc/xen/`. Este directorio contiene el demonio `xend` y otros archivos de

configuración de máquinas virtuales. Los scripts de red también se encuentran en este directorio (en el subdirectorio /scripts)

Todos los archivos de registro que pueden ser consultados durante la etapa de detección y solución de errores se encuentran en el directorio /var/log/xen.

El directorio predeterminado para todas las imágenes de discos de las máquinas virtuales basadas en archivos es /var/lib/xen.

La información del componente de virtualización para el sistema de archivos/proc está en el directorio /proc/xen.

6.2 Descripción de los archivos de registro

El componente de virtualización xen contiene el demonio xend y el proceso qemu-dm, dos utilidades que escriben varios archivos de registro en el directorio /var/log/xen:

xend.log

Es el archivo de registro que contiene todos los datos recogidos por el demonio xend, ya sean éstos eventos normales del sistema o acciones iniciadas por el operador. Todas las operaciones sobre la máquina virtual (crear, apagar, destruir, etc.) aparecen aquí. El xend.log es, generalmente, el primer lugar donde se debe buscar información para detectar problemas de rendimiento. Este archivo contiene entradas detalladas sobre los mensajes de error.

xend-debug.log es el archivo de registro que contiene entradas sobre los eventos de error de xend y los subsistemas de virtualización (framebuffer, scripts de Python, etc).

xen-hotplug-log es el archivo de registro que contiene datos sobre los eventos de conexión en caliente. Si un dispositivo o script de red no aparece en línea, el evento será registrado en este archivo.

qemu-dm.[PID].log es el archivo de registro creado por el proceso qemu-dm para cada huésped virtualizado. Cuando utilice este archivo de registro debe obtener el PID del proceso qemu-dm utilizando el comando ps. Examine los argumentos del proceso para aislar el proceso qemu-dm en la máquina virtual. Tenga en cuenta que debe reemplazar el símbolo [PID] con el PID del proceso qemu-dm.

Si encuentra algún error con el administrador de máquinas virtuales, puede revisar los datos generados en el archivo virt-manager.log ubicado en el directorio /.virt-manager . Tenga

6.3 Detección y resolución de errores

En cuenta que cada vez que inicia el administrador de máquinas virtuales, el contenido del archivo de registro será sobrescrito. Asegúrese de crear una copia de seguridad de virt-manager.log antes de reiniciar el administrador de máquinas virtuales después de un error.

Directorios importantes

Hay otras utilidades y archivos de registro que deben ser tenidos en cuenta a la hora de detectar y solucionar errores en el entorno de virtualización xen.

- Las imágenes de máquinas virtuales están en el directorio /var/lib/xen/images.
- Cuando el demonio xend reinicia se actualiza la base de datos xend-database ubicada en el directorio /var/lib/xen/xend-db.

- Los volcados de máquinas virtuales (ejecutadas con el comando `xm dump-core`) están en el directorio `/var/lib/xen/dumps`.
- El directorio `/etc/xen` contiene archivos de configuración usados para administrar recursos del sistema. El archivo de configuración del demonio `xend` se llama `xend-config.sxp` y puede ser usado para implementar cambios a lo largo del sistema y para configurar llamadas de red.

Los comandos `proc` son otro recurso para obtener información del sistema. Estas entradas `proc` están ubicadas en el directorio `/proc/xen`:

`/proc/xen/capabilities /proc/xen/balloon /proc/xen/xenbus/`

6.4 Herramientas para la detección y solución de problemas

Esta sección resume las aplicaciones para el administrador del sistema, las utilidades de red y las herramientas de depuración avanzadas. Los registros y las herramientas de administración del sistema pueden ayudar durante el proceso de detección y corrección de errores.

- `xentop`
- `xm dmesg`
- `xm log`
- `vmstat`
- `iostat`
- `lsof`

Se puede emplear estas herramientas de depuración avanzadas y registros durante el proceso de detección y solución de errores:

- XenOprofile

Herramientas para la detección y solución de problemas

- systemTap
- crash
- sysrq
- sysrq t
- sysrq w

Se puede emplear estas herramientas de red para asistirlo en la detección y corrección de errores:

- ifconfig
- tcpdump
- brctl

Brctl es una herramienta de red que inspecciona y modifica la configuración de los puentes ethernet en el Kernel de Virtualización de Linux. Debe tener acceso de root antes de ejecutar estos comandos de ejemplo:

6.5 Detección y solución de errores con la consola serial

La consola serial es de gran ayuda al detectar y solucionar problemas difíciles. Si el kernel de virtualización se cae y el hypervisor genera un error, no hay forma de rastrear el error en el host local. Sin embargo, la consola serial le permite capturarlo en el host remoto. Debe configurar el anfitrión Xen para que envíe la salida de datos a la consola serial. Luego debe configurar el host remoto para capturar esos datos. Para llevar a cabo esta tarea, modifique estas opciones en el archivo `grub.conf` para activar la consola serial 38400-bps en `/dev/ttyS0` de `com1`:

```
title Red Hat Enterprise Linux (2.6.18-8.2080_RHEL5xen0) root
(hd0,2) kernel /xen.gz-2.6.18-8.el5 com1=38400,8n1 module /vmlinuz-
2.618-8.el5xen ro root=LABEL=/rhgb quiet console=xvc console=tty
xencons=xvc
```

```
module /initrd-2.6.18-8.el5xen.img
```

CAPITULO VII

CONCLUSIONES Y ANÁLISIS.

7.1 Conclusiones

Como cierre y término de la presente monografía, este capítulo pretende hacer un breve comentario acerca del trabajo realizado mediante el uso de críticas objetivas y el señalamiento de puntos de especial interés.

Es importante remarcar que este trabajo fue por demás interesante e integral, ya que para su satisfactoria realización, fue necesario involucrar de alguna u otra manera conocimientos adquiridos a lo largo de la carrera universitaria.

Las conclusiones principales de esta tesis en orden de desarrollo son:

Como se pudo observar durante el desarrollo de la presente monografía, la técnica de virtualización con xen, permite aislar de forma segura todos los servidores que se ejecutan en un ambiente virtualizado llamados huéspedes, impidiendo de esta manera ocupar recursos en forma directa.

El entorno gráfico de administración de Xen virt-manager permite una administración intuitiva permitiendo al administrador observar en tiempo real gráficas estadísticas que indican el comportamiento del hypervisor y de cada huésped que se encuentra instalado ya sea en modo paravirtual o full-virtual.

Se puede concluir que la diferencia entre la paravirtualización y la full-virtualización se centra en el kernel, la paravirtualización modifica el kernel de cada guest para permitir realizar llamadas directas al hardware “hyperllamadas”, mientras que la full virtualización es un soporte por parte del procesador con el fin de reducir la penalización al virtualizar huéspedes que no permiten modificar el kernel, los dos métodos de virtualización tienen

un objetivo común, la de reducir la penalización al momento de virtualizar huéspedes.

Con respecto a la elaboración de una guía paso a paso para la configuración, instalación y administración del Hypervisor Xen en Centos, se ha podido elaborar un esquema comprensible para el lector, destacando puntos importantes, entre ellos la penalización que sufre el hardware al virtualizar, comprendiendo que la virtualización demanda una pequeña pérdida en los recursos, definiendo así el mejor método para virtualizar servidores destacándose la paravirtualización.

Con respecto a la instalación y administración de sistemas virtualizados, se pudo comprobar que un entorno virtualizado facilita el uso y administración tanto al instalar el hypervisor como al instalar los distintos huéspedes, brindando herramientas que permite sacar provecho de las mejores características de un entorno virtualizado.

Al finalizar el análisis de las máquinas virtuales y sus distintos métodos de virtualización se puede concluir que la técnica de virtualización es una tendencia que está de moda, por ende las mejores herramientas trabajan para incrementar las características de la virtualización.

7.2 Recomendaciones

Se recomienda el uso de la técnica de virtualización de servidores, ya que ofrece la posibilidad de expansión mucho más al usar soporte de virtualización por parte de los procesadores intel o amd, incrementando más aún sus características al fusionarlo con volúmenes lógicos, que ofrecen una forma óptima en la administración de los discos físicos.

Al incrementar los servicios de una empresa se recomienda tener en cuenta la técnica de virtualización, a la hora de virtualizar escritorios ya que brinda una

administración eficiente de los clientes y nos permite reducir costos y esfuerzo al momento de administrar cada cliente; también al momento de virtualizar los servidores ya que permite aislar y administrar los recursos eficientemente

GLOSARIO

| | |
|----------|---|
| Iptables | Firewall plataforma linux |
| Selinux | Seguridad mejorada de Linux (Security-Enhanced Linux) |
| IP | Protocolo de internet (Internet Protocol) |
| LAN | Red de área local (Local Area Network) |
| PV | Volúmen Físico (Physical Volume) |
| VG | Volúmen de Grupo (Volúmen Groups) |
| LV | Volume Lógico (Logical Volume) |
| DOM0 | Dominio Cero (Domain 0) |
| DOMU | Dominio GUEST (Domain U) |
| PE | Extensión Físico (Physical Extents) |
| PAE | Extensión de dirección física (Physical Address Extensión) |
| RAID | Conjunto redundante de discos independientes (Redundant Array of Independent Disks) |
| VMM | Monitor de máquina virtual (Virtual Machine Monitor) |
| SSH | Intérprete de órdenes segura (Security Shell Client) |

| | |
|------|---|
| UDP | Protocolo de datagrama del usuario (User Datagram Protocol) |
| HTTP | Protocolo de transferencia de Hipertexto (Hypertext Transfer Protocol) |
| DHCP | Protocolo de configuración de host dinámicos (Dynamic Host Configuration) |
| NAT | Traducción a direcciones de red (Network Address Translation) |
| MD5 | Algoritmo de Resumen del Mensaje 5 (Message-Digest Algorithm 5) |
| DNS | Servidor de nombres (Domain Server) |
| NFS | Sistema de ficheros en red (network file system) |
| PID | Número que identifica un proceso (process id) |
| FTP | Protocolo de transferencia de archivos (File Transport Protocol) |

BIBLIOGRAFÍA

[RHI2007] Red Hat Inc (2007). Red Hat Enterprise Linux 5 Virtualización A reference guide for virsh, xm, vmm and xend. ref. Aug. 2007, disponible en la web [http://docs.redhat.com/docs/es-](http://docs.redhat.com/docs/es-ES/Red_Hat_Enterprise_Linux/5/pdf/Virtualization/Red_Hat_Enterprise_Linux-5-Virtualization-es-ES.pdf)

[ES/Red_Hat_Enterprise_Linux/5/pdf/Virtualization/Red_Hat_Enterprise_Linux-5-Virtualization-es-ES.pdf](http://docs.redhat.com/docs/es-ES/Red_Hat_Enterprise_Linux/5/pdf/Virtualization/Red_Hat_Enterprise_Linux-5-Virtualization-es-ES.pdf)

[RHI2010] Red Hat, Inc (2010). Red Hat Enterprise Linux 6 Virtualización Guide Guide to Virtualización on Red Hat Enterprise Linux 6. ref. May. 14 2010, disponible en la web:

<http://www.centos.org/docs/5/pdf/Virtualization.pdf>

[XUM2008] Xen(2008). Users' Manual. ref Aug. 15 2008, disponible en la web

<http://bits.xensource.com/Xen/docs/user.pdf>

[MJ2010] Medina, Jorge (2010). Administración de entornos virtuales con Xen y GNU/Linux. ref. Jun. 27 2010, disponible en la web:

http://tuxjm.net/docs/Curso_Virtualizacion_Xen/index.html

[CA2010] Cathrow Andrew (2010). Red Hat Enterprise Virtualización Tour.

[EPE2007] Perez, Ernesto (2007). Técnicas de virtualización. ref. Feb. 28 2007, disponible en la web

http://www.ecualug.org/2007/02/28/comos/1_t_cnicas_de_virtualizaci_n

[VWK2011]Virtualización. (2011, 13) de abril. Wikipedia, La enciclopedia libre.

Fecha de consulta: 03:44, marzo 30, 2011 disponible en la web:

<http://es.wikipedia.org/w/index.php?title=Virtualizaci%C3%B3n&oldid=45599462>.

[RG1974] Robert P. Goldberg (1974). "Formal Requirements for Virtualizable Third Generation Architectures".

[VWP2011] Virtualización. (2011, 13) de abril. Wikipedia, La enciclopedia libre. Fecha de consulta: 23:11, marzo 5, 2011 from <http://es.wikipedia.org/w/index.php?title=Virtualizaci%C3%B3n&oldid=45599462>.

[QMU2011] QEMU. (2011, 5) de junio. Wikipedia, La enciclopedia libre. Fecha de consulta: 23:29, marzo 5, 2011 from <http://es.wikipedia.org/w/index.php?title=QEMU&oldid=46921801>.

[BCH2010] BOCHS. (2010, 11) de diciembre. Wikipedia, La enciclopedia libre. Fecha de consulta: 23:31, marzo 5, 2011 from <http://es.wikipedia.org/w/index.php?title=BOCHS&oldid=42443045>.

[VTB2011] VirtualBox. (2011, 19) de mayo. Wikipedia, La enciclopedia libre. Fecha de consulta: 23:34, marzo 5, 2011 from <http://es.wikipedia.org/w/index.php?title=VirtualBox&oldid=46485424>.

[VTZ2010] Virtuozzo. (2010, 17) de septiembre. Wikipedia, La enciclopedia libre. Fecha de consulta: 23:35, marzo 5, 2011 from <http://es.wikipedia.org/w/index.php?title=Virtuozzo&oldid=40333381>.

[XWP2011] Xen. (2011, 1) de junio. Wikipedia, La enciclopedia libre. Fecha de consulta: 23:37, marzo 5, 2011 from <http://es.wikipedia.org/w/index.php?title=Xen&oldid=46797237>.

[KVM2011] Kernel-based Virtual Machine. (2011, 14) de abril. Wikipedia, La enciclopedia libre. Fecha de consulta: 23:40, marzo 5, 2011 desde http://es.wikipedia.org/w/index.php?title=Kernel-based_Virtual_Machine&oldid=45617785.

[VMW2011] VMware ESX. (2011, 1) de marzo. Wikipedia, La enciclopedia libre.
Fecha de consulta: 23:42, junio 5, 2011 from
http://es.wikipedia.org/w/index.php?title=VMware_ESX&oldid=44492203.

[HYP2007] HYPERVISOR. (2007,2) de mayo. Virtualizacion, fecha de consulta: 08:20,
marzo 8, 2011 desde consulta:http://www.virtualizacion.com/?page_id=8

[HYP2] Hipervisor. (2011, 27) de mayo. Wikipedia, La enciclopedia libre. Fecha de
consulta: 00:10, mayo 6, 2011 desde
<http://es.wikipedia.org/w/index.php?title=Hipervisor&oldid=46664696>.