



UNIVERSIDAD LUTERANA SALVADOREÑA

**FACULTAD DE CIENCIAS DEL HOMBRE Y LA NATURALEZA
LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN**

CATEDRA:

SISTEMA OPERATIVO DE REDES

TEMA:

FACE FINAL DE PROYECTO DE CÁTEDRA

TÍTULO:

VIRTUALIZACIÓN CON KVM

AUTORES:

APELLIDOS	NOMBRES	CARNET	PARTICIPACION
MAZANA	MANUEL ANTONIO	MM012255	50%
GÁMEZ GÁMEZ	NORIS MARBELIS	GG02110806	50%

CATEDRÁTICO:

ING. MANUEL DE JESUS FLORES

SAN SALVADOR 08 DE NOVIEMBRE 2014

INDICE

Introducción	3
Objetivos	4
1. Definición de virtualización	5
1.1 Origen de kvm	5
1.2 Arquitectura	6
2.0 Virtualización con kvm	6
Ventajas	7
2.1 Ventajas	7
2.2 Desventajas	7
3.0 Características	8
4.0 Instalación de kvm	9-18
5.0 Conclusión	19
Recomendación	19
Bibliografía	20

INTRODUCCION:

La virtualización es una de las maneras que muchas empresas pudieran utilizar ya que de este modo pueden ahorrar energía, aprovechar los recursos y pues mejorar el flujo de datos.

Todas las nuevas tecnológicas de desarrollo están llevando a cabo en kvm, ya que va hacer la solución más recomendada, soportada por varios sistemas operativos como gnu/Linux, bds, Solaris, Mac y Windows. También de mayor seguridad entre las máquinas virtuales y una configuración más amigable para los administradores, en varias distribuciones que van a cambiar a kvm por xen entre estas ya están debían, Ubuntu, rhel, centos, federa, opensuse, mandria y entre otras más.

Poder virtualizar un servidor es de mucha ventaja para la entidad en la que se desarrolle ya que permite instalar en el mismo servidor físico a varios servidores virtuales. Con la virtualización, ya no es necesario comprar un nuevo servidor cuando se quiere desplegar un nuevo entorno de trabajo.

Nuestro trabajo está enfocado a la virtualización con kvm, acrónimo de *máquina virtual basada en el núcleo* («kernel-based virtual machine».es primero que nada un módulo del núcleo que provee la mayor parte de la infraestructura que puede usar un virtualizador. además se presenta el proceso de instalación que se lleva a cabo para lograr tener este virtualización.

Objetivos:

General:

- ✓ Configuración de un servidor de virtualización donde podamos crear máquinas virtuales usando la tecnología KVM en un sistema Debian.

Específicos:

- ✓ Conocer las herramientas necesarias para un proceso de virtualización con KVM en nuestros equipos.
- ✓ Preparar las condiciones para la realización de un servidor virtual con KVM.
- ✓ Mostrar el aprovechamiento que brinda KVM en la utilización de los recursos de Hardware.

Marco teórico

1. Virtualización

La virtualización es uno de los avances más grandes de la informática en los últimos años. El término abarca varias abstracciones y técnicas de simulación de equipos virtuales con un grado variable de independencia de hardware real. Un servidor físico puede almacenar varios sistemas que funcionan de forma simultánea y aislada. Sus aplicaciones son muchas y generalmente surgen de este aislamiento: entornos de prueba con diferentes configuraciones o separar los servicios provistos entre diferentes máquinas virtuales por seguridad.

Hay múltiples soluciones de virtualización, cada una con sus ventajas y desventajas. Este libro se concentrará en Xen, LXC y KVM; pero otras implementaciones notables incluyen las siguientes:

Qemu es un emulador en software para un equipo completo; su rendimiento está lejos de la velocidad que uno podría conseguir si ejecutara nativamente, pero esto permite ejecutar en el hardware emulado sistemas operativos sin modificación o experimentales. También permite emular una arquitectura de hardware diferente: por ejemplo, un sistema *amd64* puede emular una máquina *arm*. QEMU es software libre.

1.2 Origen de KVM

KVM se inició como un proyecto Open Source por la empresa israel Qumranet. Curiosamente se inició como un proyecto centrado en una solución VDI para clientes Windows.

Qumranet fue adquirida en 2008 por Red Hat. Red Hat ha centrado su estrategia en KVM para definirlo como la mejor solución de virtualización en el mundo Open Source.

En RHEL 5 (Red Hat Enterprise Linux), la solución de virtualización escogida era Xen, en RHEL 6 se abandonó por KVM.

KVM se implementa como un módulo en el kernel de Linux. Se integró en la rama principal del kernel de Linux en la versión 2.6.20, en el año 2007. Actualmente es el hipervisor de virtualización oficial del kernel de Linux.

IBM también es una de las empresas que apuesta fuerte por KVM.

Ha contribuido en áreas como la gestión de memoria, mejoras en el rendimiento y el subsistema de E/S virtual. IBM tiene a un gran número de desarrolladores trabajando sobre KVM y lo ofrece como solución en su portfolio de software

para servidores. Otras distribuciones de Linux también comenzaron a invertir en KVM e incluirlo en sus distribuciones como el caso de SUSE o Ubuntu.

KVM se incluyó oficialmente en la rama principal del kernel en la versión 2.6.20 (lanzada en febrero de 2007). Implicaciones:

KVM forma ahora parte integral del kernel de Linux, por lo que aparece en todas las distribuciones ya que todas llevan un kernel más actual que el 2.6.20. Que forme parte y tenga capacidades, no implica que haya soporte comercial ni que se incluyan las herramientas necesarias.

KVM es capaz de aprovecharse de la comunidad en torno a Linux, cualquier mejora sobre el kernel, es una mejora que beneficia a KVM. Cualquier desarrollador de Linux puede también beneficiarse del uso de KVM.

1.3 Arquitectura

- ✓ KVM es un módulo cargable del kernel que permite al SO Linux actuar un hipervisor "bare metal" tipo 1.
- ✓ Un hipervisor puede verse como un SO especializado en la ejecución de máquinas virtuales.
- ✓ Pero un hipervisor tipo 1 también tiene que tratar con otras tareas más estándares como gestión de memoria, planeación de procesos, drivers de dispositivos, E/S.

2.0 Virtualización con KVM

KVM, acrónimo de *máquina virtual basada en el núcleo* («Kernel-based Virtual Machine»), es primero que nada un módulo del núcleo que provee la mayor parte de la infraestructura que puede usar un virtualizador, pero no es un virtualizador en sí mismo. El control real de la virtualización es gestionado por una aplicación basada en QEMU. No se preocupe si esta sección menciona programas quemu, continúa hablando sobre KVM.

A diferencia de otros sistemas de virtualización, se integró KVM al núcleo Linux desde el comienzo. Sus desarrolladores eligieron aprovechar el conjunto de instrucciones de procesador dedicados a la virtualización (Intel-VT y AMD-V), lo que mantiene a KVM liviano, elegante y no muy hambriento de recursos. La contraparte, obviamente, es que KVM funciona principalmente en procesadores i386 y amd64, y sólo en aquellos suficientemente recientes para tener estos conjuntos de instrucciones. Puede verificar si tiene uno de estos procesadores

si encuentra a «vmx» o «svm» entre las opciones de CPU («flags») enumeradas en /proc/cpuinfo.

Con Red Hat respaldando activamente su desarrollo, KVM parece haberse convertido en la referencia de virtualización en Linux.

2.1 Ventajas y desventajas

1.1 Ventajas:

- ✓ Incluido en la rama principal del kernel de Linux. Instalación prácticamente nula.
- ✓ Cuenta con todas las ventajas que le proporciona ser parte integral del kernel de Linux.
- ✓ KVM es un hipervisor ligero, de alto rendimiento y bajo coste.
- ✓ Gran soporte.
- ✓ Listo para su uso en entornos en producción.
- ✓ Modelos de seguridad avanzado proporcionado por SELinux.
- ✓ Soporte de invitados Windows, Linux, Android, Familia BSD (OpenBSD, FreeBSD, NetBSD), Solaris, etc.

1.2 Desventajas:

- ✓ Proyecto muy joven.
- ✓ No hay herramientas sofisticadas para la gestión de servidores y para la gestión/creación de máquinas virtuales.
- ✓ KVM aún puede mejorar mucho más en áreas como: soporte de redes virtuales, soporte de almacenamiento virtual, seguridad, alta disponibilidad, tolerancia a fallos, gestión de energía, soporte
- ✓ HPC/tiempo real, etc.

3.0 Las características principales de KVM son:

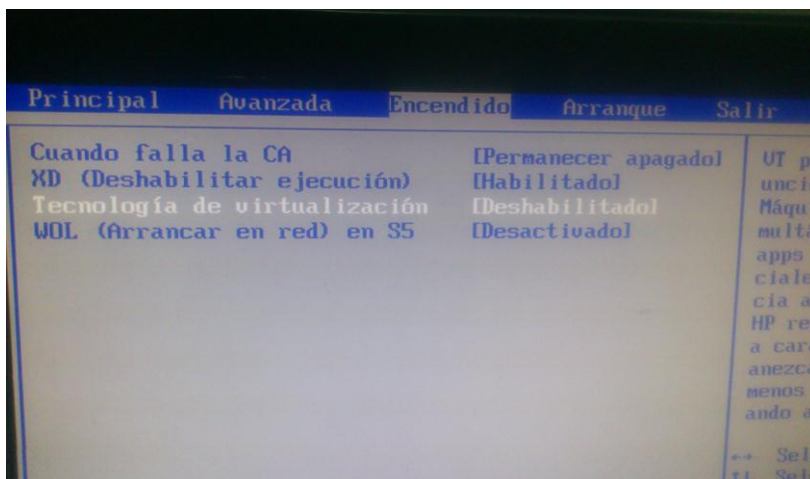
- ✓ **SEGURIDAD.** Supongamos que necesitas instalar un servidor LAMP (Linux, Apache, MySQL, PHP, esto es un servidor para montar páginas web) y un servidor de archivos puede optar por dos opciones. Instalar en un único servidor ambos servicios, LAMP y servicio para compartir archivos. Todo te funcionaría bien, ¿pero que ocurriría si un hacker encontrara una vulnerabilidad en tu web y pudiera tomar el control del servidor? Que no solo tendría acceso a los archivos de tu web, sino que también a todos los archivos que tienes compartidos, con lo cual desde un fallo en la web se ha comprometido todos los archivos de tu empresa. En cambio si se hubiera escogido una solución virtualizada se habría podido crear en el servidor 2 máquinas virtuales. En la primera se instalaría el servidor LAMP, y en la segunda se instalaría el servidor de archivos. **En este escenario el hacker que consiga encontrar una vulnerabilidad en la web solo tendría acceso a la máquina virtual** que aloja el servidor LAMP. Y aunque habría conseguido comprometer tu web los archivos de tu empresa aún seguirían aislados y seguros. Es por este motivo por el que para muchos administradores la solución ideal es crear una máquina virtual por cada servicio que tenga un riesgo medio/alto de ser vulnerado.

- ✓ **COPIAS DE SEGURIDAD.** La mayoría de los servicios que utilizamos hoy en día disponen de mecanismos para hacer copias de seguridad y poder restaurarlas en caso de fallos. Pero también es cierto que las copias de seguridad a veces no son completas, contienen datos pero no guardan configuraciones o perfiles específicos. Si utilizamos máquinas virtuales, nosotros **podemos programar en el servidor de forma automática que cada día haga una copia de la máquina virtual**, de modo que si queremos restaurar una copia de seguridad bastará con arrancar la máquina que tenemos copiada y funcionará sin tener que tomar ninguna medida más.

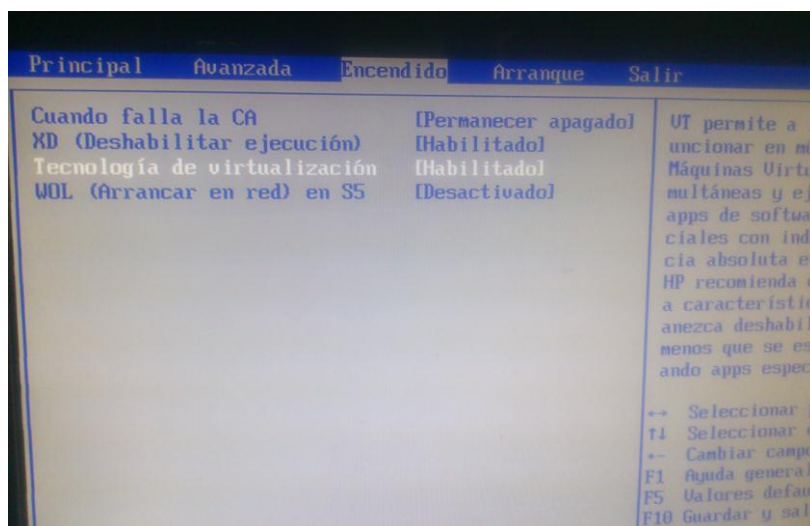
- ✓ **PROTECCIÓN CONTRA ERRORES HARDWARE.** Uno de los problemas que más daño pueden crear a un administrador de sistemas es que falle un componente crítico del servidor, la placa base por ejemplo. Si esto ocurre y hay que sustituir este elemento habrá que reinstalar todo el sistema operativo, configurarlo, instalar los programas, configurarlos y cargar las copias de seguridad para poder seguir trabajando. Todo esto tiene un coste en tiempo y esfuerzo bastante alto. Sin embargo si estamos trabajando con máquinas virtuales **si falla el servidor podremos sustituirlo, copiarle las máquinas virtuales y arrancarlas**. Esta solución no te llevará ni una décima parte de tiempo que si tuvieras que reconfigurarlo todo.

4.0 Instalación de sistema debían en KVM

Antes de empezar la instalación y configuración con KVM es necesario decir que tenemos que activar en el BIOS de nuestra computadora la opción de Virtualización, ya que por defecto siempre viene desactivado.



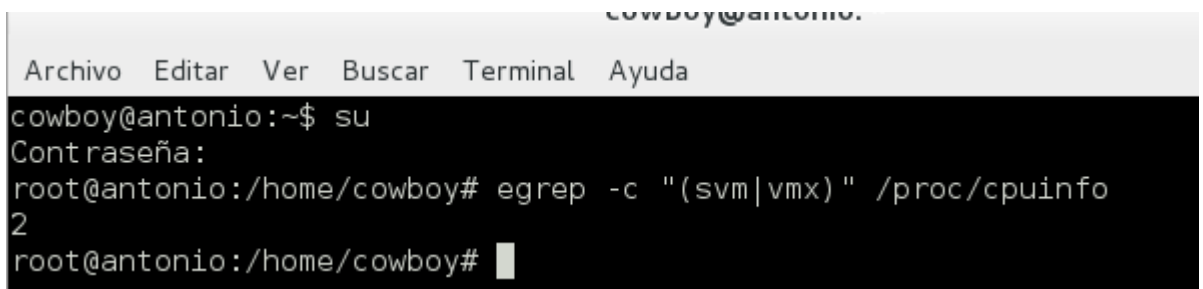
Y la dejamos de la siguiente manera y al salir guardamos los cambios y listo podemos empezar a instalar con KVM cualquier sistema operativo que queramos.



Lo primero que tenemos que hacer es ver si nuestra computadora soporta una virtualización,

KVM o **Kernel-based Virtual Machine**, funciona solamente si el procesador tiene las extensiones de virtualización de hardware, tanto las de Intel© como las de AMD-V©. Lo podemos determinar si ejecutamos en una consola el siguiente comando:

```
egrep -c "(svm|vmx)" /proc/cpuinfo
```



```
cowboy@antonio.  
Archivo Editar Ver Buscar Terminal Ayuda  
cowboy@antonio:~$ su  
Contraseña:  
root@antonio:/home/cowboy# egrep -c "(svm|vmx)" /proc/cpuinfo  
2  
root@antonio:/home/cowboy#
```

Si el comando nos devuelve 0, indica que el procesador no soporta virtualización de hardware. Un valor 1 o superior nos indicará cuantos procesadores tienen esa característica. Debemos habilitar (o comprobar si está habilitada) ésta función en la BIOS de nuestro equipo, aunque el valor devuelto sea 1 o mayor.

Ahora que ya sabemos que nuestra computadora soporta la virtualización nos disponemos a realizar la instalación de los paquetes que necesitamos.

```
apt-get install qemu-kvm libvirt-bin bridge-utils virt-manager hal
```

```
cowboy@antonio: ~
Archivo Editar Ver Buscar Terminal Ayuda
root@antonio:/home/cowboy# aptitude install qemu-kvm libvirt-bin virtinst kvm virt-viewer
```

Aclaremos que es en una estación de trabajo porque el paquete **virt-manager** es la interfaz gráfica o GUI para la administración de nuestras máquinas virtuales en nuestro Desktop y para el resto de los servidores de nuestra LAN que tengamos dedicados al soporte de máquinas virtuales. Además, el proceso de creación y administración de las máquinas o servidores virtuales se puede realizar en modo consola solamente sin la necesidad de una interfaz gráfica.

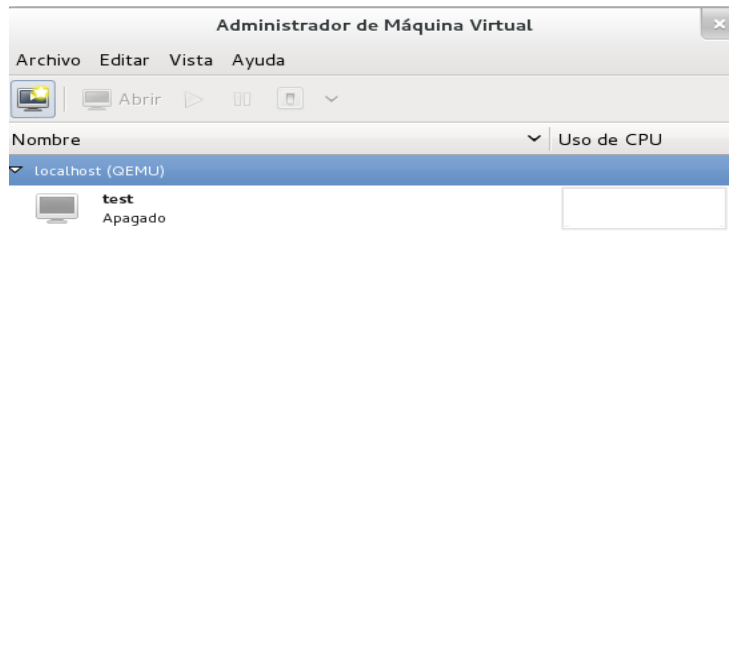
El usuario **root** y los miembros del grupo **libvirt** son los únicos que tienen permisos para utilizar las máquinas virtuales KVM. Por tanto debemos hacer a nuestro usuario miembro del grupo **libvirt**:

```
adduser miusuario libvirt
```

```
cowboy@antonio: ~
Archivo Editar Ver Buscar Terminal Ayuda
root@antonio:/home/cowboy# adduser cowboy libvirt
Añadiendo al usuario `cowboy' al grupo `libvirt' ...
Añadiendo al usuario cowboy al grupo libvirt
Hecho.
root@antonio:/home/cowboy#
```

Después de lo anterior queda poco por explicar pues ya podemos ejecutar el “Administrador de Máquina Virtual” o **virt-manager**, al que encontraremos en el grupo de aplicaciones “**Herramientas del Sistema**”, o podemos invocarlo si presionamos Alt + F2 y escribimos en la caja de diálogo **virt-manager**.

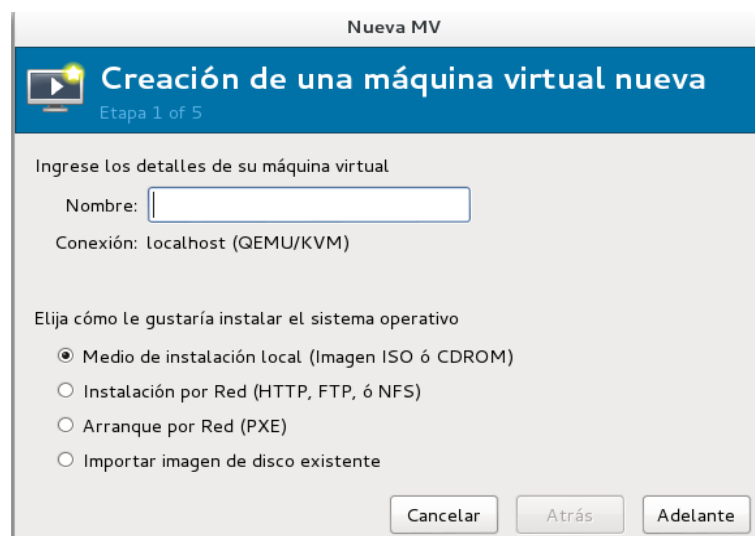
Al tener ya instalado KVM solo buscamos el administrador de máquina virtual, Nos saldrá un panel en el cual veremos las máquinas virtuales que ya tenemos instaladas o configuradas.



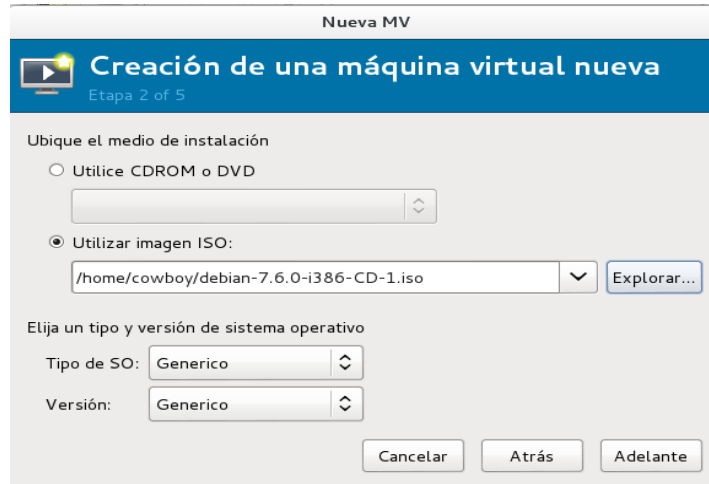
Para crear una máquina virtual nueva le damos click al icono, crear una máquina virtual nueva.



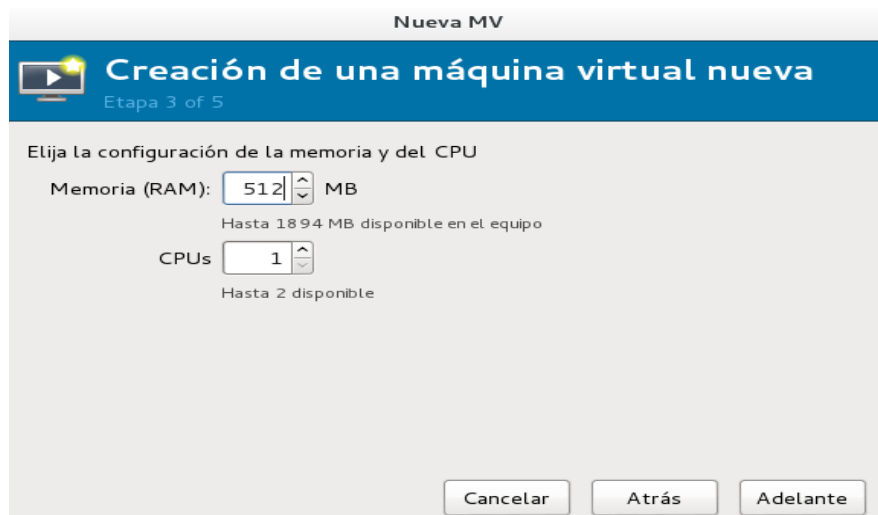
Luego nos mostrara una ventana donde pondremos el nombre de la maquina virtual que vamos a crear y el medio de instalación que utilizaremos, en mi caso es una imagen iso que tengo en mi disco-duro, luego de esto damos click en el botón Adelante.



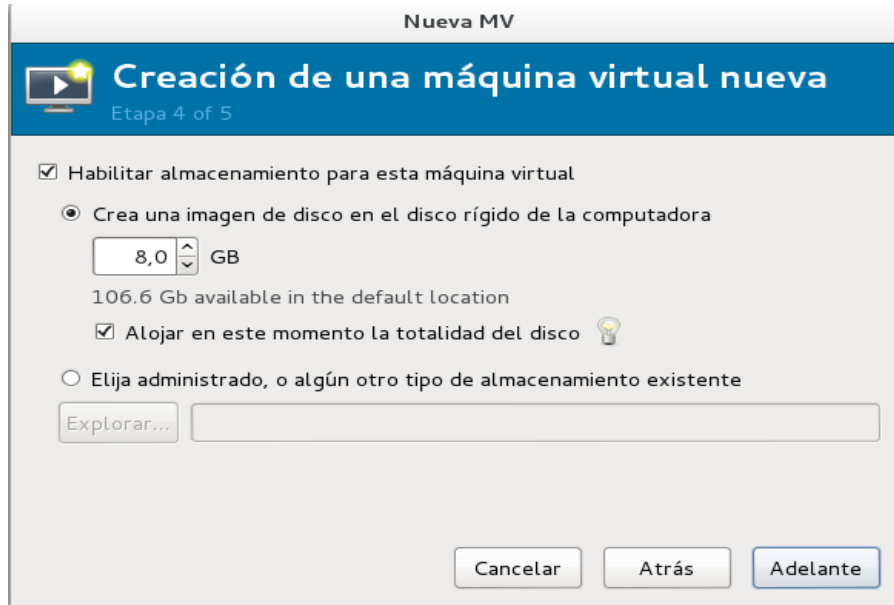
Indicamos la dirección mediante el botón "Explorar" y el Tipo de sistema y versión en los botones justamente debajo.



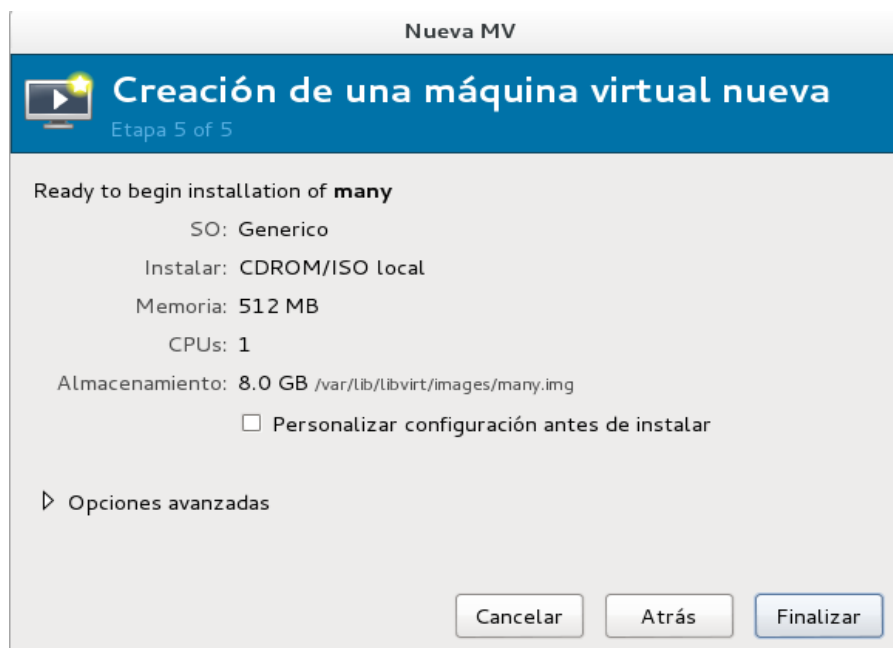
Seleccionamos la RAM y el número de núcleos que usara la máquina virtual: normalmente se asigna un núcleo por máquina virtual si se cuenta con varios núcleos se le puede asignar más núcleo y así poder crear más virtualizaciones dentro de una Virtualización, y así también en la cantidad de RAM a utilizar, luego de esto damos click en Adelante.



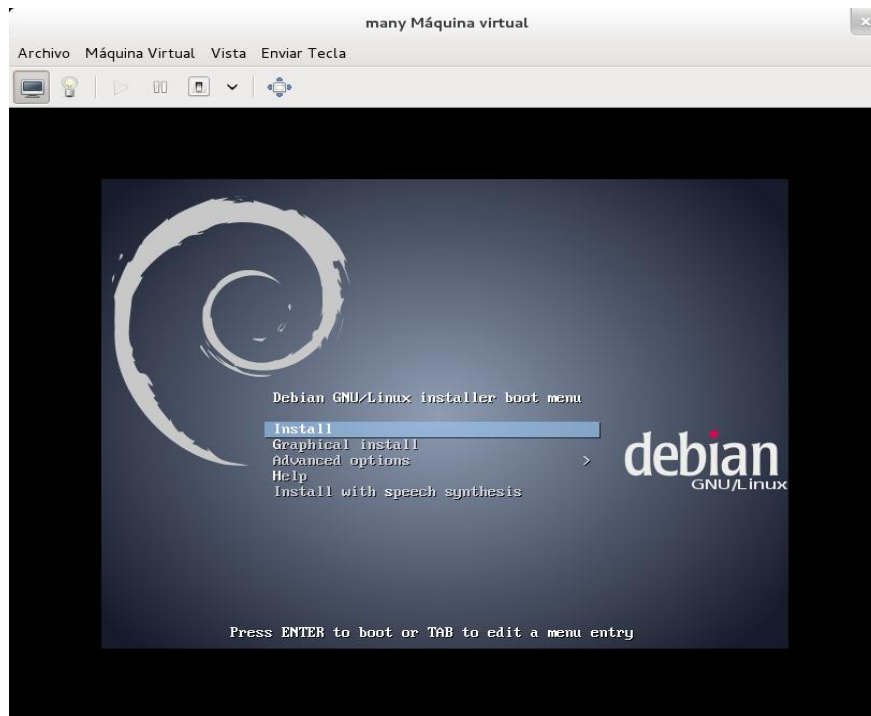
Luego indicamos la cantidad de GB que utilizara nuestra máquina virtual, y damos click en Adelante.



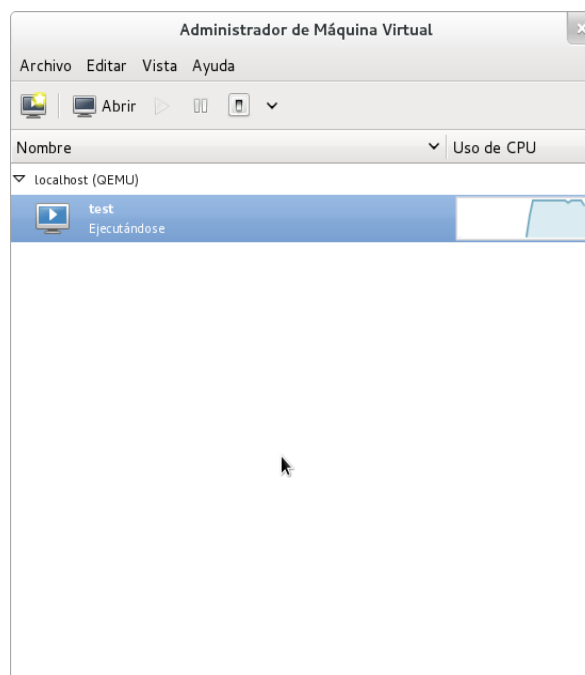
Finalmente, nos saldrá un breve resumen de la máquina, si está todo correcto, pulsamos en "Finalizar".

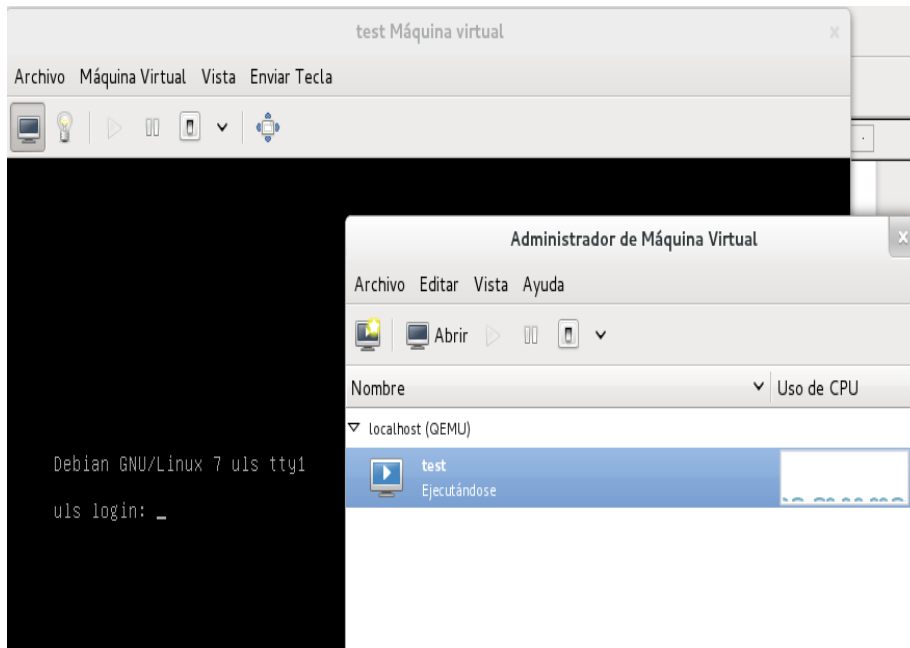


Luego nos aparece una ventana donde podemos ver que el sistema esta listo para su instalación en nuestra máquina virtual.



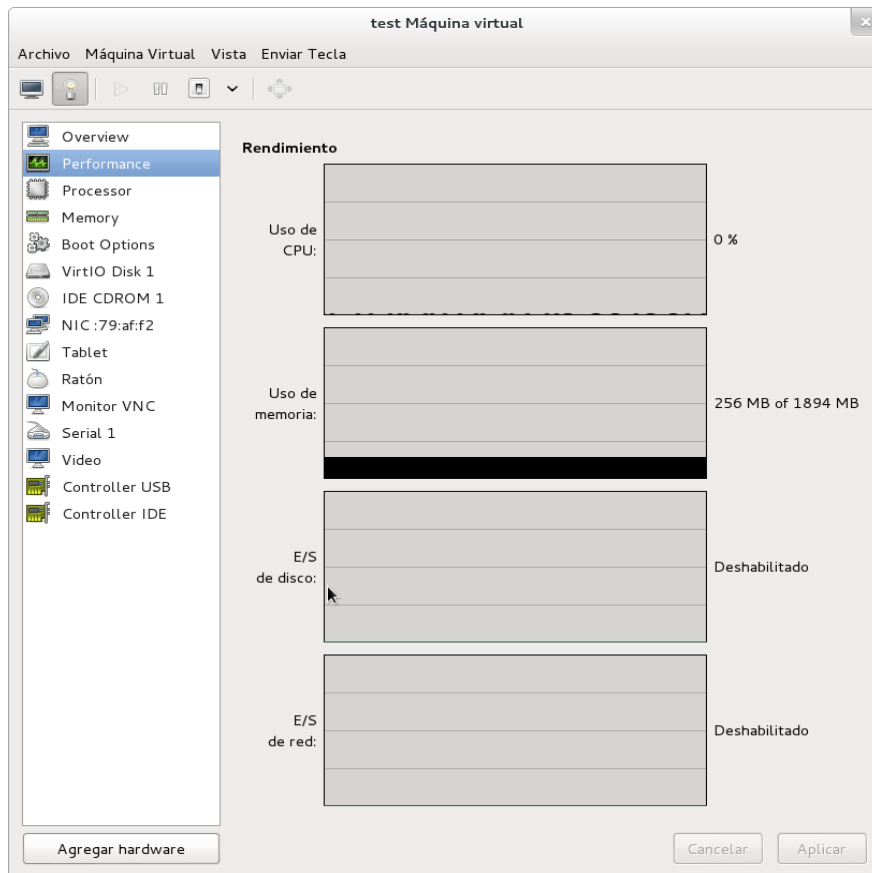
Para encender nuestra máquina virtual de damos click derecho sobre el nombre de la máquina virtual y elegimos ejecutar y veremos que empieza la actividad de nuestro equipo virtual en la parte derecha aparece el uso del CPU.



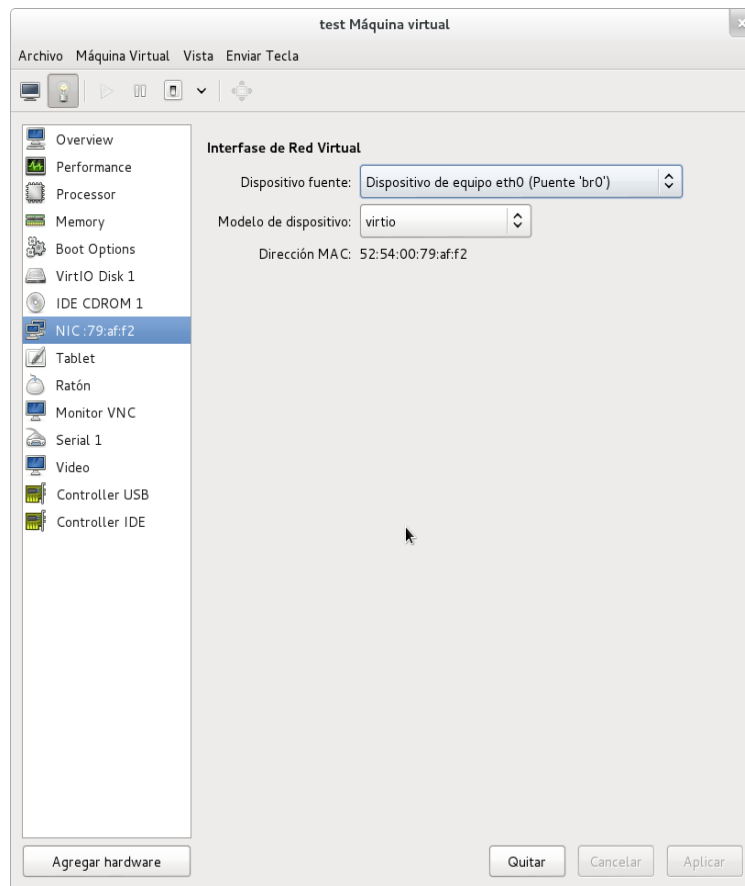


Luego damos click donde dice Abrir y nos mostrara nuestra máquina activa

en el icono que es en forma de foco podemos ver todo el hardware virtualizado que posee nuestra máquina y si queremos podemos agregar o quitarle.



Vemos la configuración de la interfase de red que usaremos en mi caso es la br0 para que las máquinas virtuales usen los rangos de IP de mi red local, para uso interno.



Como resultado de la culminación de este proyecto es importante que se tomen algunas sugerencias que serían de mucha ayuda.

- ✓ Debe de ser perseverante a la hora de realizar pruebas
- ✓ Sea observador de fallos y posible solución.

Conclusión:

No queda duda que KVM que se completa con la versión de qemu es una de las herramientas de virtualización buenísima que podemos utilizar para aprovechar las características con las que cuenta, hoy en día la virtualización es una de las mejores opciones con las que contamos pues nos ofrece una serie de ventajas que nos ayudan tanto en lo económico como en la seguridad. Trabajar en entornos virtuales cada vez se está volviendo una necesidad de manera que podemos emular hardware que quizá la empresa o en entidad en la que operemos no tenga sin olvidar que kvm también es una herramienta libre.

Recomendaciones:

Para poder empezar la instalación primero que nada debes de saber si tu pc soporta software de virtualización eso lo puedes conseguir con este comando.
egrep -c "(svm|vmx)" /proc/cpuinfo.

Además debes de configurar la red para poder instalar.

Bibliografía

URLs de referencia:

- <http://www.howtoforge.com/how-to-set-up-xen-4.3-on-debian-wheezy-7.0.2-and-then-upgrade-to-jessie>
- <http://www.howtoforge.com/paravirtualization-with-xen-4.0-on-debian-squeeze-amd64>
- http://openvz.org/Installation_on_Debian
- <http://www.howtoforge.com/virtualization-with-kvm-and-openqrm-5.1-on-debian-wheezy-p4>