

“Virtualización de servidores Linux usando Xen”

Por: Manuel Dávila Sguerra

Virtualización de servidores usando Xen

1. Introducción

Entre los años 60 y 70, los computadores eran grandes máquinas, en tamaño, que manejaban procesos centralizados de la empresas a costos tan altos que sólo se adquirirían a nivel corporativo y era impensable imaginar que se adquirieran a nivel personal.

Resumen

El crecimiento de los sistemas de información en las empresas obliga a la utilización de servidores para su operación. Se ha observado que por motivos de seguridad, privacidad y rendimiento se tiende a poner soluciones informáticas en servidores separados. Con el tiempo las empresas terminan con centros de cómputo que albergan muchos servidores cada uno, con necesidades propias de administración pero que en general se utilizan en bajos porcentajes de su rendimiento máximo.

Esta situación, denominada “Server sprawl”, que significa algo así como “servidores desperdigados” aumenta los costos de administración y crea procesos incontrolables de crecimiento en la infraestructura de servidores.

Este artículo trata el tema de una de las tecnologías de virtualización de servidores, llamada Xen, que permite instalar en un servidor varios sistemas operacionales, logrando mejorar el factor de uso de los recursos computacionales y bajar los costos, tanto en administración como en licenciamiento de software

Palabras claves:

Virtualización, Xen, rendimiento, servidores, Linux, sistemas operacionales

Con la aparición de los microcomputadores, en 1980, se inició una etapa en la cual el computador se convirtió en una opción viable para el uso personal; pero, además, con el desarrollo de los componentes de hardware y de software, fue posible elevarlo al nivel de servidor central dando auge a un proceso de descentralización de los procesos.

Esta descentralización fue acogida y defendida con argumentos relacionados con el mejoramiento en la seguridad de la información, un aumento del rendimiento de los equipos y una independencia, en caso de fallas.

Esto fue creando una práctica que en inglés se llama “Server sprawl”, es decir “servidores desperdigados”, creando una situación de exceso en la distribución de los procesos que llevó a las empresas a adquirir varios servidores, casi bajo la norma de uno por aplicación, dando lugar un incremento en los costos de funcionamiento debido a que cada uno de ellos requiere mantenimiento y administración de las aplicaciones, además de los incrementos de infraestructura física para localizarlos.

Si bien el exceso de distribución crea una situación beneficiosa para el rendimiento de los procesos crea, igualmente, un problema de costo beneficio ya que se obtiene una subutilización de los equipos. Es el caso en que esta utilización de estos equipos llegue a ser sólo el 10% o 15% de sus capacidades, determinadas estas por el uso de la C.P.U., de los dispositivos de entrada y salida y el tráfico en la red.

Lo anterior justifica estudiar el concepto llamado "virtualización de servidores" lo que ha dado motivo para la creación de productos basados en proyectos de investigación y desarrollo de muchos años que en realidad se remontan a los años 60's y 70's con los computadores como el IBM 370, que ya hablaba de máquinas virtuales.

Este concepto se convierte en una realidad mediante el desarrollo de una capa de software que, al ser instalada dentro del sistema operacional de un servidor central, permite instalar sobre ella varios sistemas operacionales que pueden ser de diferente origen. Esto aumenta el factor de uso de los recursos computacionales dentro de la misma máquina y disminuye el número de equipos que conforman la batería de servidores, lo cual disminuye las necesidades de administración de los mismos y permite una mejor distribución del software licenciado con la correspondiente baja en los costos de funcionamiento.

El cambio se da entonces si se logra pasar de un factor de uso de los recursos de un 15% a un 85% o 90%, cuidando de no sobrepasar sus capacidades para no caer en el lado contrario es decir una sobreutilización.

En este artículo trataremos uno de los estándares que ha dado el software libre llamado "El proyecto Xen"¹ que es una máquina virtual desarrollada por la Universidad de Cambridge, con el fin de ejecutar instancias de diversos sistemas operativos sin restarle funcionalidades ni características a ninguno de ellos.

2. Los primeros conceptos de máquinas virtuales

En los años setenta IBM diseñó una serie de "mainframes", denominados IBM 370, dentro de los cuales ya era posible incluir máquinas virtuales, entendiéndose el término como software no como hardware. Los computadores se construían con los recursos tradicionales de hardware como la C.P.U., la memoria, los discos duros y los protocolos de red. Sobre ellos se instalaba el sistema operacional Labor, que duraba cerca de 24 horas en comparación de hoy en día que no pasa de tres o cuatro horas.

Sobre el sistema operacional de base se instalaban entonces otros sistemas operacionales con capacidad de manejar multitareas y multiusuarios, dejando ver como si cada uno de ellos tuviera control único de la máquina procesando diversas tareas simultáneamente.

La Figura N° 1 muestra un esquema de estas características.

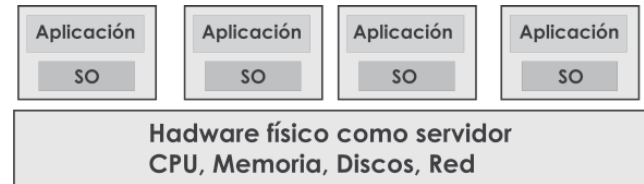


Figura N° 1. Esquema de máquinas virtuales

3. Historia de XEN

XEN es un proyecto originado por una investigación del System Research Group del Laboratorio de computadores de la Universidad de Cambridge, iniciado en el año 2001, cuya liberación al público se llevó a cabo tres años después, en 2003. Este proyecto fue liderado por Iann Prat, uno de los miembros de la Facultad del laboratorio de la Universidad de Cambridge, UK, cuya labor fundamental era la de investigador de sistemas, durante 8 años, en Ciencia de los Computadores.



Figura N° 2. Iann Prat

Esta tecnología es la base de la estrategia llamada SaaS o "Software as a service", la cual es un modelo de liberación de software, desarrollado por Tim O'Reilly quien fue el fundador de la editorial O'Reilly Media, impulsor del software libre y uno de los autores del concepto Web 2.0 quien mezcló sus intereses en la literatura y la informática desde que estudiaba en Harvard, en 1975.



Figura N° 3. Tim O'Reilly

1. Xen <http://www.cl.cam.ac.uk/research/srg/netos/xen/>

Esta es una las razones por las cuales veremos a Microsoft interesado en este proyecto que nació del software libre ya que su nuevo paradigma de comercialización del software es precisamente el SaaS. El proyecto recibe el apoyo de varias multinacionales de la informática como son: Xensource, IBM, Intel, Hp, Novell, Bull, Ncsc, Fujitsu, Redhat, Amd, Virtualiron, Cam, Unisys, Microsoft, Citrix.

Podría decirse que, soluciones como estas, se están convirtiendo en el puente entre el software propietario y el software libre.

4. Características de Xen

Durante el artículo y como parte de la investigación aplicada, que se ha desarrollado, se intentará ir introduciendo algunos términos que conforman los elementos de su diseño.

Xen se define entonces como un monitor de máquinas virtuales, identificado alternativamente como VMM o Hypervisor, diseñado para arquitecturas de procesadores de 32 y 64 bits de fuente abierta con capacidad para soportar procesadores x86 de 32 bits, x86 de 64 bits, con extensiones de direcciones físicas PAE (Physical Address Extension). Xen logra que sobre el mismo hardware se ejecuten varios sistemas operacionales al mismo tiempo, de manera segura y eficiente, con un rendimiento cercano al de un sistema operacional nativo.

Soporta hardware con drivers de Linux, hasta 32 C.P.U. virtuales en cada máquina invitada (vCPU) con funcionalidades para la asignación dinámica de recursos (hot plug) y Procesadores Intel-VT y AMD-V.

Estos productos de software pertenecen a unas familia llamadas XenServer: XenEnterprise, XenServer, XenExpress

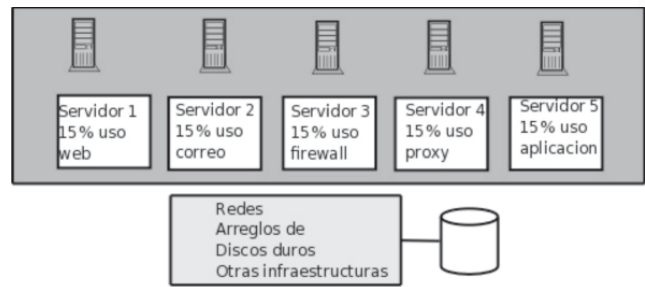
5. Descentralización vs. centralización

Una manera de concebir un proceso actual descentralizado, de servidores en una empresa se expresa en la figura Nº 4, en la cual aparecen varios servidores actuando en diferentes tareas de la empresa compartiendo una batería de discos duros como elemento común.

ver figura 4

Veamos un caso en el cual se muestran los costos y rendimientos de un sistema descentralizado² que sirve de ejemplo para la justificación de la virtualización

2. David William, 2007, Virtualization with Xen, Editorial SYNGRESS



de los servidores, como se ve la tabla en la que *Figura Nº 4. Servidores descentralizados*

se consigan los datos de costos unitarios y totales de la misma.

COMPONENTE	COSTO UNITARIO DESCENTRALIZADO EN DÓLARES	COSTO TOTAL DESCENTRALIZADO EN DÓLARES
Hardware	\$ 7.500	\$ 37.500
Licencias de software / cpu	\$ 2.000	\$ 20.000
Soporte	\$ 2.500	\$ 12.500
Energía por año	\$ 180	\$ 2.700
Calefacción por año	\$ 150	\$ 2.250
Costo total 3 años		\$ 74.950

Tabla Nº 1. Ejemplo de costos con servidores descentralizados.

La versión centralizada de esta configuración se expresa gráficamente en la Figura Nº 5 como un solo servidor, que tiene "dentro de sí" varios de ellos haciendo las tareas de manera simultánea.

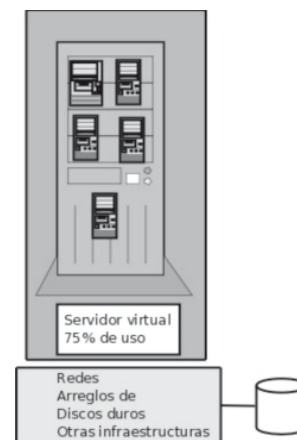


Figura Nº 5. Servidores virtuales. Centralización

Paralelamente el mismo autor presenta los costos de los diferentes componentes para este tipo de configuración de los servidores.

COMPONENTES	COSTO UNITARIO	COSTO UNITARIO DESCENTRALIZADO EN DÓLARES	COSTO VIRTUAL EN DÓLARES
Hardware	\$ 7.500	\$ 37.500	\$ 7.500
Licencias de Software / CPU	\$ 2.000	\$ 20.000	\$ 4.000
Soporte	\$ 2.500	\$ 12.500	\$ 2.500
Energía por año	\$ 180	\$ 2.700	\$ 540
Calefacción por año	\$ 150	\$ 2.250	\$ 450
Costo total 3 años		\$ 74.950	\$ 47.990

Ahorro \$ 59.960

Tabla N° 2. Ejemplo de costos con servidores virtuales

6. Arquitectura de Xen

Tal como se acostumbra a expresar gráficamente los diseños de los sistemas operacionales, como es el caso de Unix, se usa la simbología de anillos concéntricos. Esto se observa en la Figura N° 6.

Los anillos más centrales representan módulos del software con acceso no restringido es decir que tienen todos los permisos para administrar el hardware y el software que debe gobernar, mientras que las capas externas sólo pueden interactuar a través de los anillos privilegiados. El anillo central o anillo cero es el administrador del recurso de memoria principal, cobrando una responsabilidad de altísima importancia en el desempeño del sistema.

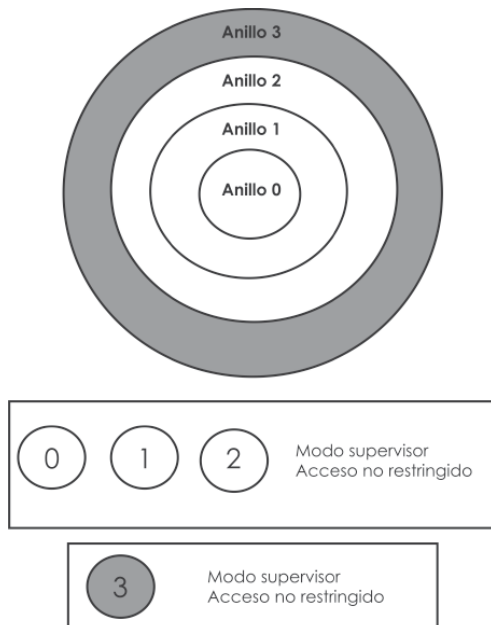
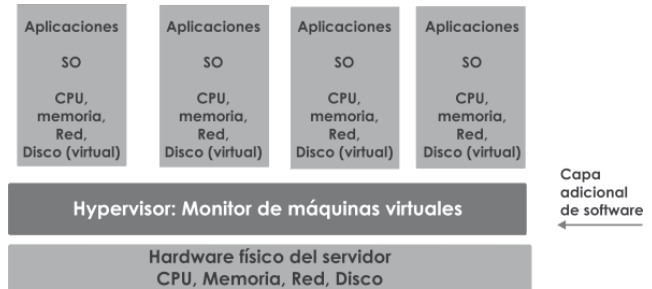


Figura N° 6. Modos de operación de Xen

6.1 Monitor de máquinas virtuales

El monitor de máquinas virtuales, o vmm, es la parte del software que administra el hardware del servidor y que a través de su intervención le presenta a los diferentes sistemas operacionales instalados en la misma máquina, que se denominan "invitados", un conjunto de recursos como la C.P.U., la memoria y los discos.



Sistema operacional y aplicaciones administradas por la capa de software del Monitor de máquinas virtuales: vmm

Figura N° 7. Monitor de máquinas virtuales

Los diferentes sistemas operacionales (SO) manejan los recursos, de manera virtual, en el sentido en que existen para ellos pero los accesos que se hagan vienen a ser manipulados, sobre los recursos reales, por el Hypervisor.

6.2 Tipos de virtualización

Se han definido dos tipos de virtualización llamados: virtualización completa y para virtualización.

La primera simula el hardware completamente dando un rango muy amplio de soporte a los diferentes sistemas operacionales con la ventaja de aislar la máquina virtual y el Hypervisor, sin tener que modificar a los sistemas operacionales, de tal manera que su rendimiento es equivalente al nativo. Requiere, sin embargo, tener una combinación precisa de los elementos de hardware y de software, con dificultades para instalarla en la arquitectura X86. Requiere atrapar y emular algunas de las interrupciones, lo cual puede bajar un poco el rendimiento.

La segunda hace una simulación parcial del hardware, dejando que cada máquina virtual tenga su propio espacio de direcciones. Su implementación se facilita y ofrece un alto rendimiento en la red y en el manejo de las interrupciones de entrada - salida. Tiene el inconveniente que el sistema operacional se debe modificar y las máquinas virtuales no son totalmente portables.

6.3 Estructura

6.3.1 Virtualización completa

Este modelo de virtualización (Figura N° 8) cuenta con una serie de APIs o interfaces de programas de aplicación, que actúan en una capa llamada micro núcleo para la administración de los requerimientos de los servidores virtuales. Estos harán solicitudes de uso de la red, procesos de enrutamiento y de acceso a los dispositivos de bloques como es el caso de los discos duros, a través de una serie de interfaces que se dividen según la funcionalidad. Estas son:

- VIF: Virtual Network Interfaces.
- VFR: Virtual Firewall y Routers.
- VBD: Virtual Block Devices.
- ACL: Access control lists para control de acceso hacia los dispositivos por parte de los anfitriones.

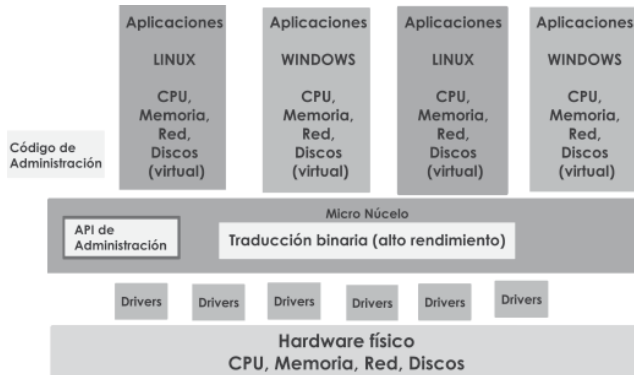


Figura N° 8. Virtualización completa

Como se puede observar en la figura los drivers atienden de manera directa la comunicación con el hardware físico.

6.3.2 Paravirtualización

Pare entender más a fondo la estructura de Xen el proceso de abstracción define, con el uso de una terminología especial, al Hypervisor como una capa de abstracción que contiene APIs o interfaces de programas de aplicación, los cuales son partes del software que ayudan a ofrecer un servicio requerido por el sistema operacional para conectarse al hardware. En la Figura N° 9 se observan más claramente las máquinas virtuales, esta vez identificadas con el sistema operacional específico a los cuales se les llama dominios y que se identifican según sean anfitriones (domU) o administradores del ambiente (dom0).

Ver figura 9

Observar cómo los anfitriones tienen drivers virtuales mientras que el dominio 0 tiene drivers reales, pasando las solicitudes de manejo del hardware al Hy

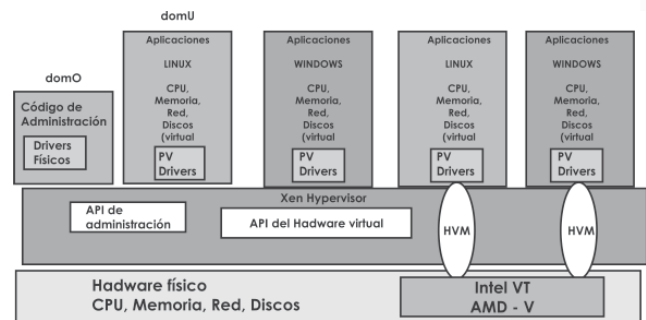


Figura N° 9. Paravirtualización

pervisor el cual tiene algunos API's involucrados dentro del software permitiendo todo esto la intervención con el hardware real.

Profundizando aun más en la estructura de la paravirtualización, la figura muestra cómo los dominios disponen de drivers llamados Frontend drivers que interactúan con los Backend drivers del dominio y del código de administración, siendo estos dos los que contienen los drivers físicos que van a manejar el hardware real.

El tráfico entre los Frontend drivers y los Backend drivers se mueve como por un "corredor" llamado canal de eventos por medio del cual se encadenan, punto a punto, usando un mecanismo de mensajes asíncronos. Esto permite que se pueda cancelar un dominio domU sin tener que cancelar el dominio 0, evitando así que fallas de los sistemas anfitriones hagan que la máquina se "caiga".

Hay un código llamado IOMMU que traslada direcciones de memoria del espacio de entrada/salida a espacio físico.

7. Instalación de Xen

El proceso de instalación para esta investigación aplicada se basó en un sistema operacional Fedora Linux, sobre el cual fue necesario instalar una serie de módulos que conforman el núcleo de Xen antes de intentar montar servidores virtuales.

Estos módulos se encuentran en el repositorio oficial de Fedora y usando el programa "pirut" se bajan al servidor Linux. Ellos son:

- kernel-xen. Núcleo de Linux compilado con Xen vm.
- xen. Monitor de máquinas virtuales.
- xen-libs. Biblioteca para las herramientas Xen.
- virt-manager. Administrador de máquinas virtuales para crear dominios o anfitriones.

Una vez instalado el núcleo de Xen, se modifica automáticamente el proceso de inicio del sistema Linux el cual es manejado por el programa llamado "grub", creando varias líneas nuevas en el proceso de arranque sobre el archivo de configuración grub.conf, como se muestra en negrillas en la Tabla N° 3.

```
# kernel /vmlinuz-version ro
root=/dev/VolGroup00/LogVol00
# initrd /initrd-version.img
default=0
timeout=5
splashimage=(hd0,1)/grub/splash.xpm.gz
hiddenmenu
title Fedora Core (2.6.20-1.2952.fc6xen)
root (hd0,1)
kernel /xen.gz-2.6.20-1.2952.fc6
module /vmlinuz-2.6.20-1.2952.fc6xen ro
root=/dev/VolGroup00/LogVol00 rhgb quiet
module /initrd-2.6.20-1.2952.fc6xen.img
title Fedora Core (2.6.18-1.2798.fc6)
root (hd0,1)
kernel /vmlinuz-2.6.18-1.2798.fc6 ro
root=/dev/VolGroup00/LogVol00 rhgb
quiet
initrd /initrd-2.6.18-1.2798.fc6.img
title Other
rootnoverify (hd0,0)
chainloader +1
```

Tabla N° 3. Grub modificado

La variable default usualmente está en 1; es decir, carga el sistema original sin Xen. Si se pone en 1 cargará el sistema con Xen.

El efecto es que al cargar el sistema con Xen automáticamente carga las librerías y el núcleo de Linux con Xen, aunque la apariencia del arranque es la tradicional excepto que al iniciar carga las librerías de Xen

7.1 Proceso de instalación:

En primer lugar se debe cargar el Administrador de máquinas virtuales gráfico llamado "virt-manager" escogiendo del interfaz de Fedora las opciones: "Aplicaciones/Herramientas del sistema/Virtual Machine Manager". El sistema pedirá la clave del administrador y mostrará una ventana, como se ve en la Figura N° 10.

La primera vez sólo aparece **Domain-0** o **Dominio 0**, que es la capa de **Xen** que está interactuando con el sistema operacional de base y los drivers.

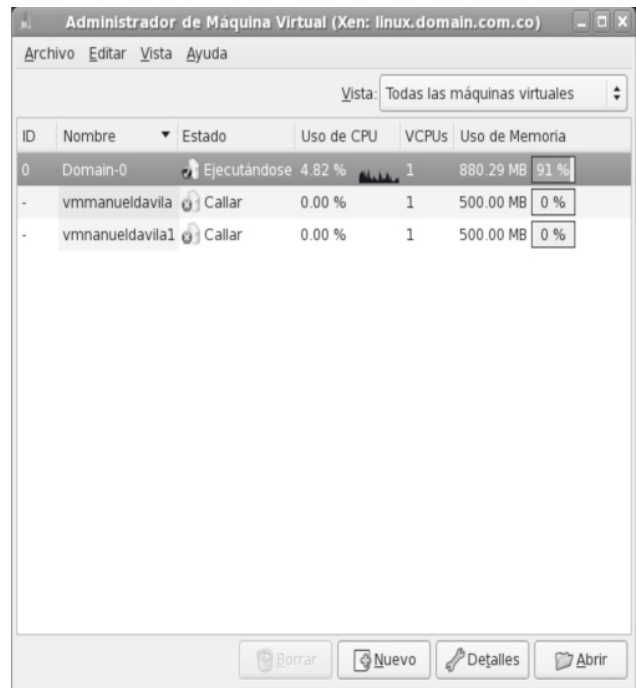


Figura N° 10. Administrador de máquinas virtuales

En este caso aparecen dos máquinas virtuales, pero se verá cómo se creó vmmanueldavila1 escogiendo "Nueva maquina", como se muestra en la Figura N° 11.

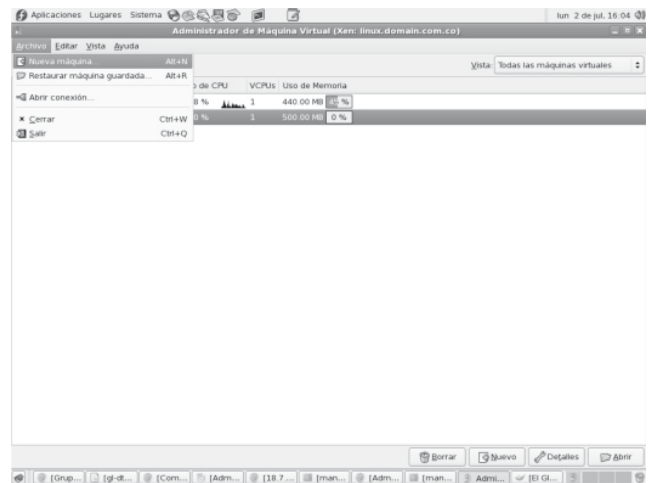


Figura N° 11. Crear nueva máquina virtual

El sistema muestra una ventana auto explicativa, Figura N° 12

Pide el nombre que se quiere para la máquina virtual, que en este caso es vmmanueldavila1 (Figura N° 13)

A continuación, solicita el método de virtualización entre Paravirtualización o Completamente virtualizado (Figura N° 14).

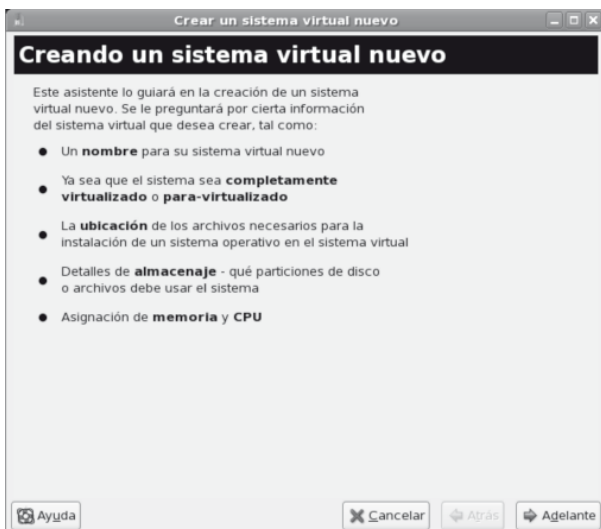


Figura Nº 12. Creando una máquina virtual



Figura Nº 13. Nombre de la máquina virtual

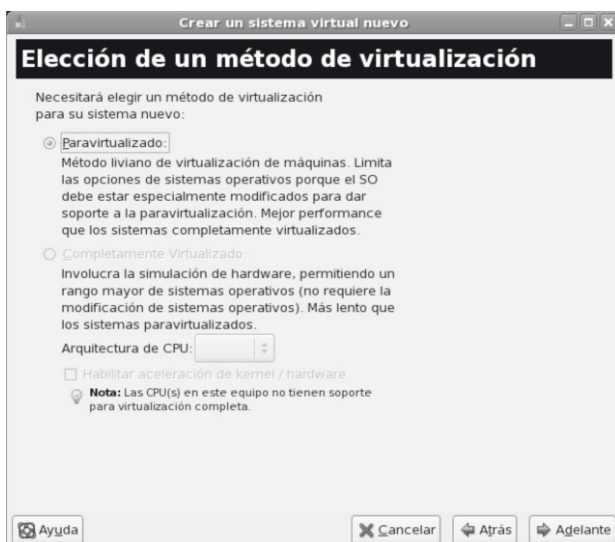


Figura Nº 14. Método de virtualización

Estando en un proceso de instalación de una nueva máquina virtual, se le debe decir al sistema en qué medio se le va a presentar. Este puede ser el repositorio oficial de Fedora 6: <http://download.fedora.redhat.com/pub/fedora/linux/core/6/i386/os/> o el directorio del servidor de web del Linux de base que contiene una copia con los discos ISO de instalación como: `/var/www/html/fc6/` por ejemplo si se quiere instalar a través de http, en cuyo caso la url de instalación sería: http://linux.domain.com.co/fedora7/F-7-i386-DVD.iso_FILES.

Esta información se le da al sistema cuando aparece esta ventana (Figura 15):

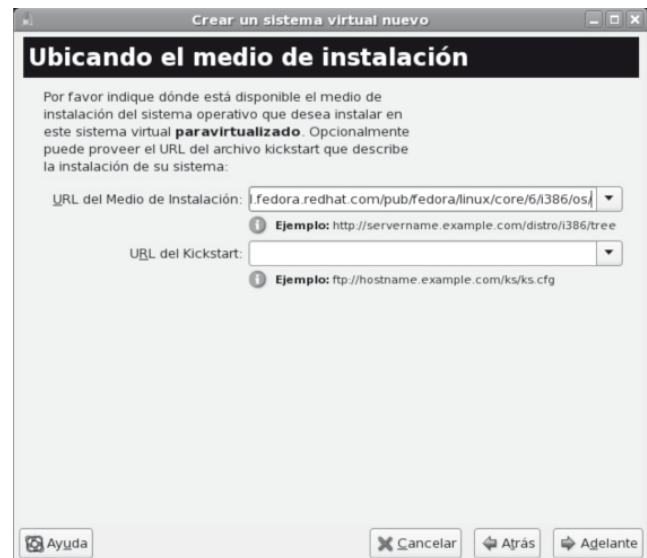


Figura Nº 15 - Medio de instalación

El sistema requiere que se le indique en dónde debe quedar la imagen de la nueva máquina virtual. Para eso es necesario conocer en qué partición del disco duro está el sistema de base.

En el caso presente se observa primero el estado de las particiones (Tabla Nº 4):

```
[root@linux tmp]# cat /proc/partitions major minor #blocks name
3 0 40021632 hda
3 1 10241406 hda1
3 2 104422 hda2
3 3 29672055 hda3
22 64 80418240 hdd
22 65 80413326 hdd1
253 0 30703616 dm-0
253 1 1966080 dm-1
```

Tabla Nº 4. Particiones

Se determina que el sistema está en /dev/hda2 que será el dato que se coloque en la Partición de disco normal (Figura 17).

La máquina virtual, que se va a instalar, se ubicará en algún directorio del sistema y en este caso se ha escogido /var/tmp, por lo que el dato que se debe colocar en la Ubicación del archivo es /var/tmp/imagenvmanueldavila1

Más tarde cuando haya instalado la máquina virtual, la imagen aparecerá en ese directorio.

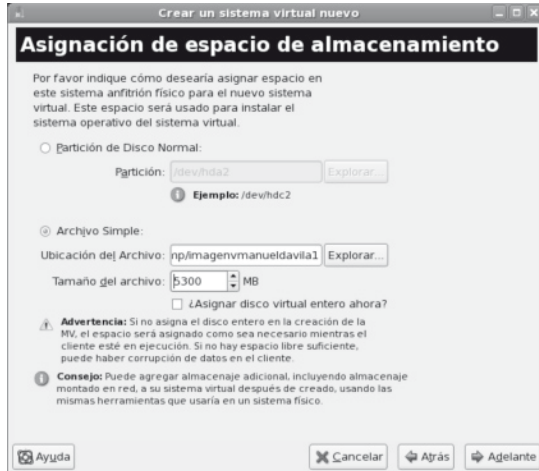


Figura Nº 16 – Asignación de espacio

La escogencia del tamaño es importante y depende de lo que se quiera instalar ahí.

El siguiente paso indica la manera como se quiere conectar la nueva máquina virtual a la red anfitriona. (Figura Nº 17).

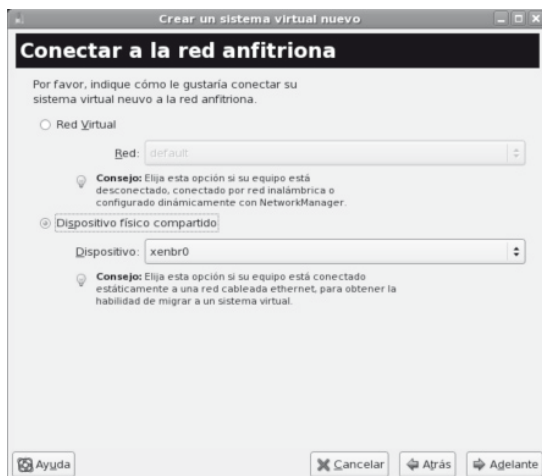


Figura Nº 17. Conectar a la máquina anfitriona

Se asigna memoria y C.P.U., como se observa en la Figura 18:



Figura Nº 18. Asignar memoria y C.P.U.

De esta manera queda todo listo para iniciar la instalación, por lo que aparece un aviso que recopila toda la información. (Figura Nº 19):

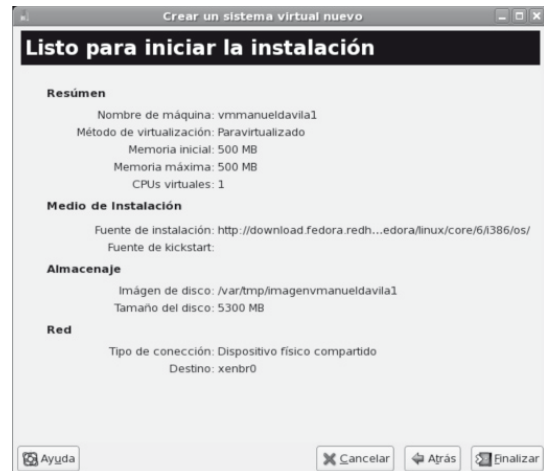


Figura Nº 19 - Recopilación de información

De esa manera se inicia la instalación de la nueva máquina virtual, en un proceso propio del sistema operacional que se esté instalando.

Cuando termina se podrá observar en /var/tmp la imagenvmanueldavila1:

```
[root@linux manuel]# cd /var/tmp/
[root@linux tmp]# ls
imagenvmanueldavila1
```

Esta máquina virtual aparecerá en el monitor como se puede observar en la Figura Nº 20:

Ver figura en la página siguiente

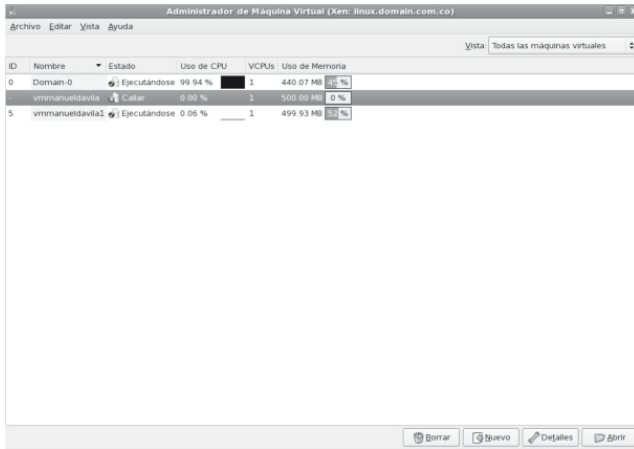


Figura N° 20 Máquina virtual activa

Automáticamente inicia la instalación del sistema operacional escogido, en este caso Fedora 6, en un proceso estándar de instalación. A partir de este momento, se tiene un doble control del ratón; uno por parte del sistema en que se está instalando y otro por parte del sistema en proceso de instalación. Si el ratón se pierde se puede hacer reaparecer usando **Ctrl-Alt**.

Se activa con el botón derecho del ratón escogiendo "Run", si se quiere cargar su sistema operacional. Al hacerlo correrá el sistema de base y el de la máquina virtual del dominio **vmmmanueldavila1**. Y se verán los dos sistemas. El de base y la nueva máquina virtual (Figura N° 21)

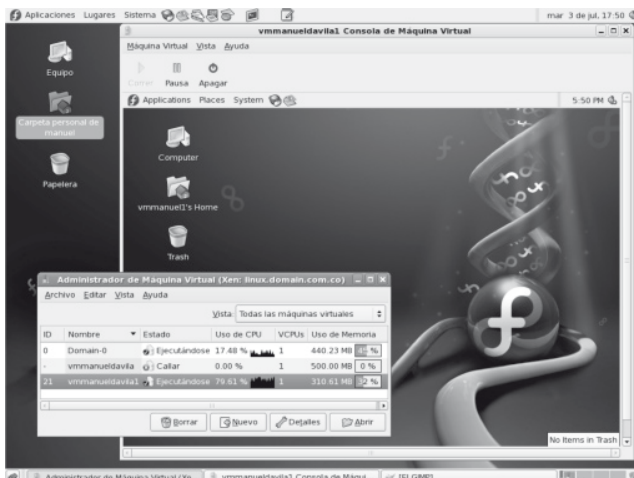


Figura N° 21. Máquina virtual en funcionamiento

8. Conclusiones

Observando el gran interés de las multinacionales de tecnología en la inclusión de las máquinas virtuales dentro de sus servidores es fácil prever que en los próximos años este tema será uno de los de más

alta competencia.

Es interesante observar cómo las tecnologías desarrolladas, en los años 70, se están convirtiendo ahora en estados del arte que comprometerán el futuro de los centros de cómputo y cómo ellas son consistentes con los estándares con los que se han desarrollado los sistemas operacionales, desde el nacimiento del sistema operacional Unix..

La literatura encontrada sobre este tema es rica en el proceso de abstracción de la arquitectura del sistema y colabora extraordinariamente en el proceso de comprensión de su diseño para facilitar su manejo, dando motivo para la creación de una terminología que es ya un estándar en el ambiente de estas soluciones y que permite dar las explicaciones más claramente.

En el proceso de su instalación, el investigador, puede entender más claramente lo que está sucediendo dentro del sistema y aunque es necesario iterar varias veces para comprender los datos que el sistema de instalación necesita, se va viendo más claramente que se está enfrente de una instalación de sistemas operacionales que, aunque operan como máquinas virtuales anfitrionas, su proceso de instalación obedece a los estándares normales como cuando se instalan de manera nativa.

Interesante observar, desde la mirada del investigador, que toda esta investigación aplicada fue posible hacerlo en equipos personales sin tener que recurrir a grandes costos en servidores profesionales, lo cual se necesitará sólo cuando se instale en equipos de aplicaciones reales en los centros de cómputo.

Esta fase primera de la investigación da inicio para una segunda en la cual se adentrará más sobre el diseño de los drivers virtuales, especialmente y a la manera cómo los sistemas operacionales anfitriones dan uso a la red para aclarar aún más el manejo y mantenimiento de las máquinas virtuales en producción, tal vez incluyendo métricas de comportamiento.

Bibliografía

Wikipedia:

<http://es.wikipedia.org/wiki/Xen>

Virtual machine manager home:

<http://virt-manager.et.redhat.com/>

Guía de virtualización:

<http://www.redhat.com/docs/manuals/enterprise/RHEL-5-manual/Virtualization-en-US/index.html>

Creando una nueva máquina virtual:

<http://www.redhat.com/docs/manuals/enterprise/RHEL-5-manual/Virtualization-en-US/virt-install-wizard.html>

Preparación para instalar una copia de discos iso:

<https://www.redhat.com/docs/manuals/enterprise/RHEL-4-Manual/es/ppc-multi-install-guide/s1-steps-hd-install.html>

Repositorio de Fedora 6:

<http://download.fedora.redhat.com/pub/fedora/linux/core/6/i386/os/>

Sitio de Xen : <http://www.cl.cam.ac.uk/research/srg/netos/xen/>

David William, 2007. Virtualization with Xen, Editorial SYNGRESS