



Tarea #4

Instalación y Configuración de Máquinas Virtuales: Windows, Kali Linux, CentOS y

Presentado a: Diego Barragan

Valentina Perez, Cod.: 2341132.

ABSTRACT:

La virtualización es una tecnología clave en la administración y prueba de sistemas operativos. Este trabajo aborda la instalación y configuración de máquinas virtuales utilizando sistemas operativos Windows, Kali Linux, CentOS y Scientific Linux. Se exploran herramientas como VMware y VirtualBox, comparando su rendimiento y compatibilidad. Se detallan los requisitos de hardware, la asignación de recursos y la configuración óptima para cada sistema operativo. Este estudio es relevante para administradores de sistemas, desarrolladores y entusiastas de la seguridad informática que buscan un entorno de pruebas controlado y eficiente.

INTRODUCCIÓN

La virtualización ha revolucionado la manera en que los sistemas operativos y aplicaciones son desplegados, probados y administrados. Gracias a herramientas como VirtualBox y VMware, es posible ejecutar múltiples sistemas operativos en una sola máquina física, maximizando el uso de los recursos y facilitando tareas como el desarrollo de software, la administración de servidores y la ciberseguridad.

Este documento describe el proceso de instalación y configuración de máquinas virtuales con cuatro sistemas operativos ampliamente utilizados: Windows, Kali Linux, CentOS y Scientific Linux. Windows es el sistema operativo comercial más utilizado, Kali Linux es una distribución especializada en ciberseguridad, CentOS es una opción robusta para servidores empresariales, y Scientific Linux es una distribución enfocada en entornos científicos.

A lo largo de este trabajo, se abordarán los requisitos de hardware y software, las configuraciones recomendadas y las mejores prácticas para optimizar el rendimiento de

cada sistema operativo en entornos virtualizados. Este estudio busca proporcionar una guía clara y estructurada para la implementación eficiente de máquinas virtuales, siendo de utilidad tanto para estudiantes como para profesionales de TI.

II. MARCO TEÓRICO

1. Virtualización:

La virtualización es una tecnología que permite la creación de entornos simulados dentro de un sistema físico, permitiendo la ejecución de múltiples sistemas operativos en una misma máquina. Existen diferentes tipos de virtualización, entre ellos la virtualización completa, la para virtualización y la contenedorización.

2. Hipervisores

Los hipervisores son el software que permite la virtualización. Se dividen en:

Tipo 1 (bare-metal): Se instalan directamente sobre el hardware del sistema, como VMware ESXi o Microsoft Hyper-V.

Tipo 2 (hosted): Se ejecutan sobre un sistema operativo, como VirtualBox y VMware Workstation.

3. Sistemas Operativos Utilizados

Windows: Sistema operativo comercial desarrollado por Microsoft, utilizado en entornos de usuario final y empresarial.

Kali Linux: Distribución basada en Debian especializada en pruebas de penetración y auditoría de seguridad.

CentOS: Sistema operativo basado en Red Hat Enterprise Linux, utilizado principalmente en servidores.

Scientific Linux: Distribución basada en Red Hat, optimizada para entornos científicos y de investigación.

4. Herramientas de Virtualización

Las herramientas más utilizadas en la virtualización incluyen:



Oracle VirtualBox: Software gratuito y de código abierto que permite la creación y gestión de máquinas virtuales.

VMware Workstation: Software comercial con un mayor rendimiento y soporte para configuraciones avanzadas.

Microsoft Hyper-V: Solución integrada en Windows para virtualización empresarial.

En este estudio, se explorarán las configuraciones óptimas para cada sistema operativo, garantizando un rendimiento adecuado según los recursos disponibles en el sistema anfitrión.

III.

IV. PROCEDIMIENTO Y RESULTADOS

A. Windows

1. se muestra el proceso de verificación de la virtualización en el equipo. Este paso es fundamental para asegurarse de que la CPU admite la ejecución de máquinas virtuales. Se ejecuta el comando 'lscpu' o 'grep -E -color 'vmx|svm' /proc/cpuinfo' en la terminal para verificar si la virtualización está habilitada en el procesador.

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

B. Verificación de KVM

Esto se logra con el comando 'kvm-ok', el cual indica si el sistema soporta KVM y si está correctamente configurado.

```
INFO: /dev/kvm exists
KVM acceleration can be used
```

C. Verificación de los servicios de libvirtd

Se verifica que el servicio 'libvirtd', necesario para administrar máquinas virtuales en QEMU/KVM, se encuentra en ejecución. Se usa el comando 'systemctl status libvirtd' para comprobar su estado.

```
● libvirtd.service - Virtualization daemon
   Loaded: loaded (/lib/systemd/system/libvirtd.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2025-03-16 14:16:58 -05; 3min ago
   TriggeredBy: ● libvirtd-admin.socket
                ● libvirtd-ro.socket
                ● libvirtd.socket
   Docs: man:libvirtd(8)
         https://libvirt.org
  Main PID: 4835 (libvirtd)
    Tasks: 19 (limit: 32768)
   Memory: 30.1M
   CGroup: /system.slice/libvirtd.service
           └─4835 /usr/sbin/libvirtd
             └─4973 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq
             └─4974 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq
```

D. Iniciar libvirtd

El comando 'sudo systemctl start libvirtd', el cual se ejecuta para iniciar el servicio 'libvirtd' en caso de que no este activo.

```
$ sudo systemctl enable --now libvirtd
$ sudo systemctl start libvirtd
$
```

E. Instalacion QEMU

Se muestra la instalación de QEMU en el sistema con el comando 'sudo apt install qemu-kvm libvirt-daemon-system libvirt-clients bridge-utils'.

```
nager virtinst libvirt-clients bridge-utils libvirt-daemon-system -y
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
bridge-utils ya está en su versión más reciente (1.6-2ubuntu1).
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  chromium-codecs-ffmpeg-extra gstreamer1.0-vaapi
  libgstreamer-plugins-bad1.0-0 libva-wayland2
  Utilice «sudo apt autoremove» para eliminarlos.
Paquetes sugeridos:
  auditd nfs-common open-iscsi pm-utils radvd systemtap zfsutils
  python3-guestfs ssh-askpass
Se instalarán los siguientes paquetes NUEVOS:
  libvirt-clients libvirt-daemon-system qemu-kvm virt-manager
  virtinst
0 actualizados, 5 nuevos se instalarán, 0 para eliminar y 27 no actualizados.
Se necesita descargar 1.473 kB de archivos.
Se utilizarán 9.218 kB de espacio de disco adicional después de esta operación.
Des:1 http://co.archive.ubuntu.com/ubuntu focal-updates/main amd64 libvirt-clients amd64 6.0.0-0ubuntu8.20 [344 kB]
Des:2 http://co.archive.ubuntu.com/ubuntu focal-updates/main amd64 libvirt-daemon-system amd64 6.0.0-0ubuntu8.20 [67,9 kB]
Des:3 http://co.archive.ubuntu.com/ubuntu focal-updates/main amd64 qemu-kvm amd64 1:4.2-3ubuntu6.30 [13,1 kB]
Des:4 http://co.archive.ubuntu.com/ubuntu focal-updates/universe amd64 virtinst all 1:2.2.1-3ubuntu2.2 [183 kB]
Des:5 http://co.archive.ubuntu.com/ubuntu focal-updates/universe amd64 virt-manager all 1:2.2.1-3ubuntu2.2 [865 kB]
Descargados 1.473 kB en 4s (365 kB/s)
Preconfigurando paquetes ...
Seleccionando el paquete libvirt-clients previamente no seleccionado.
(Leyendo la base de datos ... 279320 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../libvirt-clients_6.0.0-0ubuntu8.20_amd64.deb
```

F. Creacion del disco virtual

Se muestra el comando 'qemu-img create -f qcow2 /var/lib/libvirt/images/windows.qcow2 160G', que genera un disco virtual en formato 'qcow2' con un tamaño de 160GB.

```
f qcow2 Windows10.qcow2 60G
Formatting 'Windows10.qcow2', fmt=qcow2 size=64424509440 cluster_size=65536 lazy_refcounts=off refcount_bits=16
```

G. Creación de maquina virtual

muestra la creacion de la maquina virtual con el



comando 'virt-install --name WindowsVM --memory 8192 --vcpus 4 --disk path=/var/lib/libvirt/images/windows.qcow2,format=qcow2 --cdrom /home/juan/Sistemas Operativos/win11.iso --os-variant=win10 --network bridge=virbr0,model=virtio --graphics spice --boot uefi'.

```
4 \
> -enable-kvm \
> -cpu host \
> -m 4G \
> -smp 2 \
> -drive file=Windows10.qcow2,format=qcow2 \
> -boot order=c \
> -device virtio-net-pci \
> -usb -device usb-tablet \
> -vga qxl \
> -netdev user,id=net0 -device e1000,netdev=net0 \
> -display gtk,gl=on
qemu-system-x86_64: warning: nic virtio-net-pci.0 has no peer
```

H. Iniciar la máquina virtual

Se inicia de la máquina virtual con 'virsh start WindowsVM'.

```
> -drive file=Windows10.qcow2,format=qcow2 \
> -boot order=c -device virtio-net-pci -usb -device usb-tablet \
> -vga qxl -netdev user,id=net0 -device e1000,netdev=net0 -display gtk,gl=on
qemu-system-x86_64: warning: nic virtio-net-pci.0 has no peer
^Cqemu-system-x86_64: terminating on signal 2
```



B. Kali

1. Creación disco virtual

Se utiliza el comando 'qemu-img create -f qcow2 /var/lib/libvirt/images/kali.qcow2 60G', que genera un disco virtual en formato 'qcow2' con un tamaño de 60GB.

2. Creación de la máquina virtual

Este comando configura la memoria, los procesadores y el disco de almacenamiento: 'virt-install --name KaliVM --memory 4096 --vcpus 2 --disk path=/var/lib/libvirt/images/kali.qcow2,format=qcow2 --cdrom /home/usuario/Descargas/kali-linux.iso --

osvariant=debian10 --network bridge=virbr0,model=virtio --graphics spice --boot uefi'.

3. Instalación de Kali Linux



5. Scientific linux

Actualmente, la imagen ISO de Scientific Linux ya no está disponible para su descarga en los sitios oficiales del ' proyecto. Esto se debe a que los desarrolladores decidieron migrar sus esfuerzos a CentOS Stream, una alternativa más moderna y compatible con las necesidades de investigación científica y computación avanzada.

V. CONCLUSIONES

La virtualización es una herramienta esencial en la administración de sistemas operativos, ya que permite ejecutar múltiples entornos en una misma máquina física sin comprometer su estabilidad. A través de este estudio, se detalló el proceso de instalación y configuración de máquinas virtuales utilizando Windows, Kali Linux, CentOS y Scientific Linux, destacando sus características, requerimientos y configuraciones óptimas.

El uso de hipervisores como VirtualBox y VMware facilita la gestión eficiente de estos entornos, proporcionando flexibilidad y escalabilidad para diferentes propósitos, desde el desarrollo de software hasta la ciberseguridad y la administración de servidores. Además, cada sistema operativo evaluado cumple funciones específicas dentro del ecosistema informático, lo que permite a los usuarios elegir la opción más adecuada según sus necesidades.



En conclusión, la implementación de máquinas virtuales no solo optimiza el uso de los recursos de hardware, sino que también permite un entorno seguro y controlado para pruebas y aprendizaje. Con una correcta configuración y asignación de recursos, la virtualización se convierte en una solución eficiente para la investigación, el desarrollo y la administración de infraestructuras tecnológicas modernas.

VI. REFERENCIAS

- I. Oracle Corporation. (2023). *Oracle VM VirtualBox User Manual*. Recuperado de <https://www.virtualbox.org>
- II. VMware, Inc. (2023). *VMware Workstation Pro Documentation*. Recuperado de <https://www.vmware.com>
- III. Microsoft Corporation. (2023). *Introduction to Hyper-V on Windows*. Recuperado de <https://learn.microsoft.com/en-us/windows-server/virtualization/hyper-v>
- IV. Offensive Security. (2023). *Kali Linux Documentation*. Recuperado de <https://www.kali.org/docs>
- V. CentOS Project. (2023). *CentOS Linux User Guide*. Recuperado de <https://www.centos.org>
- VI. Scientific Linux. (2023). *Scientific Linux Overview*. Recuperado de <https://scientificlinux.org>



UNIVERSIDAD SANTO TOMÁS
PRIMER CLAUSTRO UNIVERSITARIO DE COLOMBIA

FACULTAD INGENIERÍA ELECTRÓNICA

