

Práctica 1: Instalación de la plataforma de virtualización KVM. Creación y configuración de una máquina virtual

Objetivos: El primer objetivo de esta actividad consiste en realizar la instalación de la *plataforma de virtualización KVM*, que será el *hipervisor* que emplearemos para la creación de toda la infraestructura virtual necesaria para desarrollar las prácticas de la asignatura. El segundo objetivo es probar la infraestructura instalada mediante la creación e instalación de una primera máquina virtual empleando la herramienta *virt-manager*, una utilidad con una interfaz gráfica que permite la administración de entornos virtuales KVM. Para la consecución de estos objetivos, aprenderemos a configurar los servicios requeridos. Con ello avanzaremos en la creación de la infraestructura virtual necesaria para desarrollar las prácticas de la asignatura.

1 Introducción

Como hemos indicado, dos son los objetivos de esta actividad práctica. El primero de ellos consiste en realizar la instalación de la plataforma de virtualización KVM, que será el hipervisor que emplearemos para la creación de toda la infraestructura virtual necesaria para desarrollar las prácticas de la asignatura. El segundo objetivo consiste en realizar la instalación de una primera máquina virtual empleando la herramienta *virt-manager*, una utilidad con una interfaz gráfica que permite la administración de entornos virtuales KVM. Información detallada relacionada con esta práctica la podemos encontrar en las siguientes fuentes de documentación electrónica:

- “*Virtualization Getting Started Guide*” [1]. En esta fuente electrónica se proporciona una introducción a conceptos e ideas básicas sobre virtualización, así como una descripción de los principales componentes de la plataforma de virtualización de *Red Hat*.
- “*Virtualization Deployment and Administration Guide*” [2]. En esta fuente se describen todos aquellos aspectos relevantes relacionados con la instalación de la plataforma de virtualización *Red Hat* en un sistema anfitrión *Linux Red Hat Enterprise*, así como la instalación de máquinas virtuales en el sistema anfitrión. Además, también cubre todos los tópicos relativos a la administración del sistema anfitrión, almacenamiento, y dispositivos, así como la administración de las máquinas virtuales usando distintos recursos de la plataforma de virtualización *Red Hat* (*virsh*, *virt-manager*, *qemu* y *libvirt*). Esta fuente es la base de esta guía.

- “*Fedora Server Documentation. Virtualization*” [3]. En esta fuente se describe cómo instalar, en sistemas que tienen instalada la distribución *Fedora Server*, el entorno de virtualización requerido para las actividades prácticas de la asignatura. De esta fuente se han extraído algunos detalles de configuración que son específicos de los sistemas de anfitriones que utilizan como sistema operativo *Fedora Server*.

2 Pasos previos

Se recomienda al estudiante hacer un breve repaso a órdenes básicas de administración de sistemas operativos Linux:

- Gestión de archivos: `mkdir`, `cp`, `mv`, `rm`, `less`, `cat`, `grep`, `find`, etc...
- Configuración básica de *SELinux* (comprobar su modo de funcionamiento – `sestatus`, comprobación de variables que definen su comportamiento – `getsebool/setsebool`).
- Gestión de paquetes mediante la orden `dnf` (equivalente a `yum` o `apt-get`).
- Gestión de servicios mediante la orden `systemctl`.
- Configuración del nombre de un host mediante la orden `hostnamectl`.
- Gestión del estado de los recursos del sistema: `top`, `free`, `df`, `lsmod`, etc...

3 Plan de actividades y orientaciones

Fase 1. Cumplimentar la encuesta sobre datos básicos del puesto de trabajo asignado y configuración previa del sistema anfitrión

Todos los PCs del laboratorio disponen de la instalación genérica de Windows y otras instalaciones de Linux. Entre las instalaciones Linux disponibles existen dos instalaciones de Linux con la distribución *Fedora Server*, **ANFITRIÓN1** y **ANFITRIÓN2**, que serán las que emplearemos en la asignatura. Se les dará indicaciones sobre la instalación concreta que deben emplear en función del grupo de prácticas en el que cada estudiante se encuentre.

Antes de comenzar a trabajar en las actividades prácticas, se les asignará un puesto de trabajo que será el que emplearán durante todo el cuatrimestre para la realización de las actividades prácticas. Todas las actividades prácticas serán defendidas exclusivamente en el puesto de trabajo asignado del laboratorio. **La primera tarea (Tarea 1) que deberá realizar en esta primera fase de la práctica será personalizar la clave del usuario root del PC asignado.** Además, como segunda tarea (**Tarea 2**), debe completar la encuesta “Datos PC” en el curso *moodle* donde se le pedirán los siguientes datos: identificador del PC asignado (este identificador se encuentra en una etiqueta amarilla en cada PC), dirección IP del PC, clave del usuario root. **Una vez se haya informado de la clave del usuario root respondiendo a la encuesta *moodle* indicada, ésta no se podrá cambiar salvo incidencia y, en caso de cambio, se deberá comunicar al profesorado responsable del grupo de prácticas.**

Una vez realizadas las dos tareas indicadas en el párrafo anterior, antes de instalar el entorno de virtualización utilizado en las actividades prácticas de la

asignatura, debe realizar las siguientes tareas básicas de configuración en el sistema anfitrión:

- Comprobar el modo de funcionamiento de *SELinux* y asegurarse que se está ejecutando en modo “*enforcing*” (**Tarea 3**). **Es MUY IMPORTANTE asegurarse de este punto antes de realizar cualquier otra tarea en el sistema anfitrión.**
- Instalar un entorno gráfico *GNOME mínimo* (**Tarea 4**). Para ello deberá instalar primero el grupo de paquetes *@base-x* (entorno gráfico X para sistemas Unix/Linux) y a continuación los paquetes *gnome-shell* (interfaz gráfica GUI de *GNOME*), *gnome-terminal* (terminal de consola de *GNOME*) y *nautilus* (gestor de archivos GUI de *GNOME*). Este entorno gráfico nos permitirá además hacer uso de la herramienta gráfica *virt-manager* para el manejo del entorno de virtualización.
- Además, se deben instalar las siguientes herramientas: *firefox* y *gedit*. El explorador *Firefox* se utilizará para acceder a los recursos de la asignatura alojados en el campus virtual y el editor de textos *gedit* como opción GUI para la edición de archivos de configuración. Este conjunto de componentes gráficos facilitarán la operatoria durante las prácticas.
- Una vez instalados estos paquetes, debe configurar el sistema para que el modo gráfico sea por defecto el modo de funcionamiento del sistema en los siguientes arranques. Se recomienda reiniciar el sistema para su comprobación.
- Configurar el nombre del sistema (**Tarea 5**), mediante la orden *hostnamectl*, para establecer un identificador FQDN (*Fully Qualified Domain Name* – nombre de dominio completamente cualificado) siguiendo el patrón *IdentificadorPC.vpd.com*, siendo *IdentificadorPC* el identificador que figura en la etiqueta que posee el equipo que está utilizando. **Para evitar confusiones, ponga el IdentificadorPC en minúsculas.**

Fase 2. Verificar que el sistema asignado cumple con los requerimientos mínimos para desplegar un sistema anfitrión de virtualización

Requerimientos del sistema

Este apartado está dedicado a describir los requerimientos que debe cumplir el sistema anfitrión para ejecutar de una manera satisfactoria máquinas virtuales mediante *KVM*. Se asumirá que el sistema operativo instalado en la máquina anfitriona es *Fedora Server* (distribución basada en *Red Hat Enterprise Linux*). El hipervisor *KVM* para *Red Hat Enterprise Linux* (o *Fedora Server*) está disponible para máquinas con procesadores con arquitectura *Intel 64* o *AMD64*.

Los requerimientos mínimos que debe cumplir el sistema anfitrión para instalar los componentes básicos de la plataforma de virtualización *Red Hat* son:

- Procesador Intel 64 o AMD64, o
 - Sistemas basados en la arquitectura x86; Intel VT-x y Intel 64 extensions.
 - Sistemas basados en la arquitectura AMD; AMD-V y AMD64 extensions.
- 2 Gbytes de memoria principal.
- 6 Gbytes de espacio libre en disco.

La siguiente tarea que debe acometer (**Tarea 6**) consiste en verificar que los procesadores del sistema anfitrión tienen habilitada la extensión para facilitar la virtualización. Para ello se debe comprobar que en el fichero del sistema en el que se reflejan las propiedades de los procesadores del sistema se encuentra activo el flag *vmx* (para procesadores Intel) o el flag *svm* (para procesadores AMD).

Una vez verificado que se cumplen los requerimientos anteriores, se debe considerar que para que nuestro entorno de virtualización (sistema anfitrión y máquinas virtuales) funcione correctamente se deben tener en cuenta las siguientes recomendaciones:

- Un núcleo de procesador o *hyper-thread* por cada número máximo de CPUs virtuales de cada máquina virtual y uno para el host.
- 2 Gbytes de memoria RAM más una cantidad adicional RAM que dependerá de la memoria principal usada por las máquinas virtuales.
- 6 Gbytes de espacio en disco más el espacio de disco requerido por cada máquina virtual. Muchos sistemas operativos requieren un mínimo de 6 Gbytes de espacio en disco. La cantidad de espacio requerido por cada máquina virtual depende del tipo de formato utilizado en los ficheros que contienen las imágenes de los discos.
 - Formato *raw*. Para máquinas virtuales que utilizan este formato el espacio de disco será (ver ecuación 1):

$$\sum \text{Espacio requerido por los ficheros raw} + 6 \text{ Gbytes del sistema anfitrión} + \text{espacio de swap que requerirá cada máquina virtual.}$$

$$\text{total for raw format} = \text{images} + \text{hostspace} + \text{swap} \quad (1)$$

- Formato *qcow*. Este formato permite que los archivos que contienen las imágenes crezcan dinámicamente, por tal motivo se debe estimar el máximo de espacio de disco requerido por la máquina virtual (ver ecuación 2):

$$\sum \text{Espacio requerido por cada fichero qcow} * 1.01 + 6 \text{ Gbytes del sistema anfitrión} + \text{espacio de swap que requerirá cada máquina virtual.}$$

$$\text{total for qcow format} = (\text{expected maximum guest storage} * 1.01) + \text{host} + \text{swap} \quad (2)$$

Un aspecto que se debe tener en cuenta, para un correcto rendimiento del sistema anfitrión y sus máquinas virtuales, es el espacio de *swap* del sistema anfitrión. El tamaño de este espacio se determina a partir del espacio de memoria RAM del sistema anfitrión. Tradicionalmente se ha recomendado que el tamaño del espacio *swap* fuera proporcional al tamaño de la memoria RAM, en general que dicho espacio fuera el doble de la RAM del sistema. Esto era así en un contexto donde los servidores tenían una cantidad de memoria RAM bastante limitada. Sin embargo, los sistemas modernos para virtualización generalmente incluyen una gran cantidad de memoria RAM y en consecuencia se recomienda que el espacio de *swap* se calcule en función de la carga de memoria habitual del sistema [4].

En cuanto al almacenamiento, las máquinas virtuales pueden usar las siguientes opciones:

- Archivos almacenados en medios locales.
- Particiones de disco.
- Volúmenes SAN conectadas por protocolo *SCSI* u otros que lo encapsulan (*iSCSI*, *FC*, *FCoE*)
- Volumen *LVM*.
- Sistemas de archivos *NFS*.
- Sistemas de archivos en clúster de tipo *GFS2*.

Considerando lo expuesto, utilizando los recursos que proporciona el sistema operativo, se debe responder a las siguientes cuestiones relativas a su sistema **(Tarea 7)**:

- ¿Cuántos chips de procesador posee el sistema?
- ¿Cuántos núcleos posee cada chip?
- ¿Cuántos hilos de procesamiento se podrán ejecutar en el sistema?
- ¿Qué cantidad de memoria RAM dispone el sistema?
- ¿Qué cantidad de RAM está disponible?
- ¿Cuál es el tamaño del área de *swap*?
- ¿Cuál es el estado de ocupación y espacio libre del almacenamiento persistente?

Compatibilidad de las máquinas virtuales KVM

El documento de *Red Hat "Virtualization Deployment and Administration Guide"* contiene información para verificar si un procesador determinado soporta las extensiones de virtualización y cómo activarlas en el caso de que éstas estén desactivadas (sección *1.2 KVM Hypervisor Requirements*).

El sistema operativo *Red Hat Enterprise Linux 7* (en nuestro caso *Fedora Server 39*) posee una serie de limitaciones referentes al número máximo de procesadores y cantidad máxima de memoria soportada. *Red Hat Enterprise* proporciona información detallada y específica de estas limitaciones para:

- Sistemas anfitriones: <https://access.redhat.com/articles/rhel-limits>
- Límites con KVM: <https://access.redhat.com/articles/rhel-kvm-limits>

Un listado de los sistemas operativos certificados anfitriones que son soportados por KVM se puede encontrar en <https://access.redhat.com/articles/973133>.

Algunos de los procesadores soportados por Red Hat Enterprise Linux 7 son los siguientes:

- **Opteron_G4:** AMD Opteron 62xx (Gen 4 Class Opteron).
- **Opteron_G3:** AMD Opteron 23xx (Gen 3 Class Opteron).
- **Opteron_G2:** Opteron 22xx (Gen 2 Class Opteron).
- **Opteron_G1:** Opteron 240 (Gen 1 Class Opteron).
- **SandyBrige:** Intel Xeon E312xxx (Sandy Brige).
- **Nehalem:** Intel Core i7 9xx (Nehalem Class Core i7).
- **Penryn:** Intel Core Duo P9xxx (Penryn Class Core 2).
- **Conroe:** Intel Celeron_4x0 (Conroe/Merom Class Core 2).
- **Westmere:** Westmere LE56xx/E56 (Nehalem-C).
- **cpu64-rhel5:** Red Hat Enterprise Linux 5 supported QEMU Virtual CPU version (cpu64-rhel5).
- **cpu64-rhel6:** Red Hat Enterprise Linux 6 supported QEMU Virtual CPU version (cpu64-rhel6).

Restricciones del hipervisor qemu-kvm que viene incluido con Red Hat Enterprise Linux 7

- **Máximo número de procesadores virtuales por máquina virtual.** En *Red Hat Enterprise Linux 7.2* este límite es 240 (160 en el caso de *Red Hat Enterprise 7.0*).
- **Virtualización sobre virtualización (*Nested virtualization*).** Esta característica está soportada a partir de *Red Hat Enterprise Linux 7.2* pero en modo “*Technology Preview*”¹. Según se indica en *Red Hat*, es una especie modo de testeo sin soporte y en ningún caso pensada para poner en entornos de producción.
- **Bit TSC Constante.** Los sistemas que no posean un “Contador de Marca de Tiempo Constante” requerirán una configuración adicional en la máquina virtual. Para verificar si se dispone de este elemento, ejecutar la orden:

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

- **Memoria *overcommit*.** La memoria *overcommit* es una funcionalidad soportada por el hipervisor que permite que la suma de las memorias de las máquinas virtuales sea mayor que la memoria física del sistema anfitrión. *KVM* soporta memoria *overcommit* haciendo uso de la memoria *swap* del

sistema, pero hay que tener presente que el uso frecuente de este espacio motivará una caída en el rendimiento del sistema. Para un correcto dimensionamiento del espacio swap es recomendable tener presente las recomendaciones de *Red Hat Enterprise*.

- **CPU *overcommit*.** No se recomienda más de 10 procesadores virtuales por núcleo de procesador físico. No está soportado que en una máquina virtual estén en estado de ejecución un número mayor de procesadores virtuales que el número de núcleos de procesadores físicos existentes en el sistema anfitrión. Aunque un *hyperthread* de un procesador físico se puede considerar como un núcleo, su rendimiento varía dependiendo del escenario del sistema, por tal motivo, en general no se debería esperar un rendimiento similar al de un núcleo.
- **Dispositivos SCSI virtuales.** *KVM* en *Red Hat Enterprise Linux* no soporta la emulación de SCSI.
- **Dispositivos IDE virtuales.** *KVM* soporta por cada máquina virtual 4 dispositivos IDE virtuales emulados.
- **Dispositivos para-virtualizados.** Se trata de dispositivos de entrada/salidas virtuales puros que sólo se pueden utilizar en máquinas virtuales, estando diseñados para obtener un rendimiento óptimo en las operaciones de entrada/salida. *Red Hat Linux 7* soporta hasta un máximo de 32 slots *PCI* por máquina virtual y 8 funciones *PCI* por slot, obteniéndose así 256 funciones *PCI* por máquina virtual en el caso de que la opción multifunción *PCI* esté habilitada. Sin embargo, se debe considerar que:
 - Cada máquina virtual soporta hasta 8 funciones de dispositivo asignadas.
 - Por defecto 4 slots *PCI* están asignados a sendos dispositivos emulados. Sin embargo, se pueden eliminar dos de ellos (adaptador de video en el slot 2 y el controlador de dispositivo de memoria *balloon*). Por tanto, cada máquina virtual podrá disponer de 30 slots *PCI*.
- **Migración de máquinas virtuales.** Un dispositivo físico se dice que está asignado cuando ha sido cedido a una máquina virtual para su uso exclusivo. Como la asignación de un dispositivo físico implica el uso de hardware en el sistema anfitrión, una máquina virtual con algún dispositivo asignado no puede ser migrada a otro sistema anfitrión mientras posea dispositivos asignados. Si el sistema operativo de la máquina virtual soporta *hot-plugging*, entonces se podría desconectar los dispositivos asignados

para luego proceder a la migración o al salvaguardado/recuperación de la máquina virtual. Las migraciones en vivo sólo son posibles entre sistemas anfitriones que posean:

- Procesadores del mismo tipo (*Intel-Intel* o *AMD-AMD*).
- El bit *No eXecution (NX)* con el mismo valor; ya sea a *on* o a *off*.

Por último, para poder llevar a cabo migraciones las operaciones de escritura en los dispositivos de bloques se deben realizar sin el uso de memoria caché (*cache=none*).

- **Dispositivos de almacenamiento.** Hay que tener precaución cuando se asigna a una máquina virtual un dispositivo de almacenamiento físico (disco o partición de disco), por ejemplo `/dev/sdb` o `/dev/sdb1`. Esto se debe que podrían acceder al dispositivo concurrentemente la máquina virtual y el sistema anfitrión, pudiendo surgir un problema de integridad en los datos almacenados. Por ello se recomienda que el sistema anfitrión sea configurado para que no tenga acceso al dispositivo asignado.
- **SR-IOV.** Se debe comprobar en qué dispositivos está verificada esta funcionalidad. Ejemplo de dispositivos en los que sí está verificada son:
 - Controladora Gigabit Ethernet Intel 82576NS (*igb driver*).
 - Controladora *Gigabit Ethernet Intel 82576EB (igb driver)*.
 - Controladora *10 Gigabit Ethernet Intel 82599ES (ixgbe driver)*.
 - Controladora *10 Gigabit Ethernet Intel 82599EB (ixgbe driver)*.
- **Asignación de dispositivos PCI.** La asignación de dispositivos *PCI* a máquinas virtuales requiere que en el sistema anfitrión posea el soporte *Intel VT-d* (o *AMD IOMMU*).
- **Aplicaciones.** Máquinas virtuales con un flujo de entrada/salida alto deben utilizar dispositivos de entrada/salida paravirtualizados. Sin paravirtualización se deben evitar algunas aplicaciones con un alto flujo de entrada/salida, como por ejemplo:
 - Servidor *Kdump*.
 - Servidor *Netdump*.
- **Otras restricciones.** Un listado completo de restricciones y usos que afectan a los entornos de virtualización se describe en [2] en la sección “*C.2 Feature Restrictions*”.

Fase 3. Instalación de los paquetes de virtualización

En esta sección se describe cómo instalar los distintos módulos software que configuran un sistema anfitrión de virtualización que tiene instalado *Fedora Server*. Asumiremos que en el sistema anfitrión ya se ha instalado este sistema operativo.

Aunque el entorno de virtualización para *Fedora Server* es el proporcionado por *Red Hat*, la forma en que se distribuyen los paquetes y grupos de paquetes en ambas distribuciones difiere. En el caso de *Fedora Server*, el entorno de virtualización se distribuye mediante el grupo de paquetes llamado *Virtualization* o alternativamente, el grupo *Virtualization-headless* que es el soporte para sistemas sin entorno gráfico. Seguidamente se muestra la configuración de estos:

Nombre del grupo de paquetes	Descripción	Paquetes obligatorios o predeterminados	Paquetes opcionales
<i>Virtualization</i>	Proporciona el entorno completo de virtualización	virt-install, libvirt-daemon-config-network, libvirt-daemon-kvm, qemu-kvm, virt-manager, virt-viewer	guestfs-tools, python3-libguestfs, virt-top
<i>Virtualization-headless</i>	Entorno de virtualización que no requiere entorno gráfico para su ejecución.	virt-install, libvirt-daemon-config-network, libvirt-daemon-kvm, qemu-kvm, systemd-container	guestfs-tools, python3-libguestfs, virt-top

La siguiente tarea que debe realizar es la instalación completa del grupo de paquetes del entorno de virtualización KVM (grupo de paquetes denominado *Virtualization*), la configuración de los servicios requeridos y la comprobación de que la instalación y configuración realizada es correcta (Tarea 8). Por instalación completa se entiende la instalación de todos los paquetes (predeterminados, obligatorios y opcionales) del grupo de paquetes *Virtualization*.

Una vez instalado el entorno de virtualización y como una primera comprobación de que la instalación se ha realizado correctamente, se puede ejecutar la orden `virsh` para obtener información acerca de los procesadores soportados por las máquinas virtuales. Para ello se debe utilizar la orden de la siguiente manera:

```
# virsh cpu-models arch
```

Donde `arch` indica la arquitectura del procesador. Por ejemplo, para la arquitectura `x86_64`, la orden sería:

```
# virsh cpu-models x86_64
```

El listado completo de modelos de CPUs soportado se encuentra en el directorio cuya ruta es `/usr/share/libvirt/cpu_map`.

Una vez instalado el grupo de paquetes **Virtualization** con todos sus paquetes (predeterminados, obligatorios y opcionales), se debe configurar el sistema para que el servicio **virtnetworkd** se inicie durante el arranque del sistema (**Tarea 9**). Este servicio permite que todas las funcionalidades relacionadas con los recursos de red estén disponibles, independientemente de que sean requeridos por los recursos virtuales existentes en el sistema. Una vez realizada esta configuración, con el objeto de verificar que ya tenemos el sistema anfitrión preparado para abordar las restantes prácticas de la asignatura, se debe reiniciar el sistema y una vez que el sistema se haya reiniciado, realizar las siguientes comprobaciones:

- Que el servicio **virtnetworkd** se está ejecutando.
- Que los módulos del núcleo **kvm** y **kvm_intel** se encuentran cargados. Estos módulos del núcleo son los elementos básicos del hipervisor KVM. Para realizar esta comprobación debe utilizar la orden `lsmod`.

Seguidamente se describen los componentes básicos del entorno de virtualización instalado. Instalando solo estos componentes se tendría un sistema anfitrión con funcionalidades básicas en relación al entorno de virtualización KVM, pero como ya se ha expresado, en nuestro caso se requiere la instalación completa del entorno de virtualización.

Paquete	Descripción
<code>qemu-kvm</code>	Emulador <i>KVM</i> a nivel de usuario en el sistema anfitrión.
<code>libvirt-daemon-config-network</code>	Ficheros de configuración por defecto para configurar redes NAT.
<code>libvirt-daemon-kvm</code>	Librerías para el servidor que permiten la interacción con el hipervisor y sistemas anfitriones. En este paquete está incluido el demonio <code>libvirtd</code> responsable de las invocaciones a las funciones de la librería, manejo de las máquinas virtuales y controla al hipervisor.
<code>virt-install</code>	Paquete que incluye la orden <code>virt-install</code> para crear máquinas virtuales desde la línea de órdenes.
<code>guestfs-tools</code>	Conjunto de herramientas que se pueden utilizar para ejecutar tareas por lotes sobre los huéspedes, obtener estadísticas de disco usado/libre, realizar copias de seguridad y clonar huéspedes, cambiar información de registro/UUID/nombre de host, construir huéspedes desde cero, etc.

virt-manager	Herramienta gráfica para administrar máquinas virtuales, (requiere el paquete libvirt-client)
--------------	---

Un servicio que se deberá utilizar en varias prácticas es el servicio *NFS*. Por ello, se debe comprobar que la configuración de *SELinux* permite que el entorno de virtualización (dominio *SELinux* denominado *virt*) puede hacer uso del servicio *NFS*. Esta comprobación se realiza obteniendo el valor de la variable booleana de *SELinux* *virt_use_nfs*. Compruebe que el valor de esta variable es “on”, lo que permitirá a los procesos que pertenezcan al dominio *SELinux* *virt* utilizar el servicio *NFS*.

Fase 4. Creación de una máquina virtual

Se trata de la última tarea (**Tarea 10**) a realizar en esta práctica y consiste en la creación de una máquina virtual e instalación del sistema operativo *Fedora Server 41* con una configuración mínima. En el proceso de creación de la máquina virtual hay que indicar qué sistema operativo vamos a instalar y cuál es la fuente de esa distribución. Tiene accesible mediante *NFS* una imagen ISO de *Fedora Server* en el servidor con dirección 10.22.146.216 (disnas2.dis.ulpgc.es) en el directorio /imagenes/fedora/41/isos/x86_64. **Es una exigencia realizar esta primera instalación empleando esta fuente y sin copiar localmente la imagen ISO de Fedora Server en el sistema anfitrión. Para las sucesivas prácticas, conviene tener copiada localmente la imagen ISO de Fedora Server 41 en un directorio local. Es importante que el directorio donde se almacene la imagen ISO no esté bajo el directorio /root, puesto que obviamente existen restricciones de acceso a ese directorio por parte de procesos lanzados en el sistema. Lo mejor es ubicarlo en un subdirectorio que esté bajo el directorio raíz del sistema /, por ejemplo, /ISO.**

Paso 1: definición de la configuración de la máquina virtual

Para la creación de la máquina virtual vamos a usar la utilidad *virt-manager*. En ella elegimos crear una nueva máquina virtual. A continuación, hay que proporcionar los siguientes parámetros.

- 1) Nombre de la máquina virtual (**mvp1**) y medio de instalación del S.O. (local).
- 2) Elección del modo de instalación: seleccionar la opción “Medio de instalación local (imagen ISO o CDRom)”. Una vez seleccionada esta opción, especificar la ruta, en su sistema anfitrión, del archivo que contiene la imagen ISO del sistema operativo a instalar. El nombre de este archivo adoptará la siguiente forma *Fedora-Server-netinst-x86_64-41-X.X.iso*. Representando el carácter X un número que identifica la versión de la distribución *Fedora Server 41* que se va a instalar.
- 3) Cantidad de memoria: 2GB.
- 4) Número de procesadores: 1.
- 5) Disco de la máquina virtual: 10 GB.

- 6) Especificación de características avanzadas de la máquina virtual: Interfaz de red, MAC, tipo de virtualización y arquitectura del procesador (1 interfaz de red en modo NAT).
- 7) Paso final: pantalla que indica el avance en la ejecución de la orden de creación de la máquina virtual especificada.

Paso 2: instalación del sistema operativo en la máquina virtual

Una vez proporcionados estos parámetros, tomará el control el instalador del S.O. Las especificaciones que deberá cumplir la instalación son las siguientes:

- 1) Particionado del disco automático.
- 2) Instalación mínima de la versión 41 de *Fedora Server*.
- 3) Habilitar la cuenta de administración *root* y permitir el acceso *SSH* a la máquina virtual al usuario *root*. Esto permitirá administrarla desde el host anfitrión mediante una conexión *SSH*.

Paso 3: configuración del sistema operativo instalado en la máquina virtual

Una vez finalizada la instalación del S.O. de la máquina virtual, reiníciela y realice los siguientes pasos de configuración en el S.O. de esta máquina:

- 1) Instalar el servicio **qemu-guest-agent** y configurar el sistema para que se inicie en los siguientes arranques.
- 2) Establecer como nombre del sistema el identificador FQDN **mvp1.vpd.com**. Para ello se debe utilizar la orden **hostnamectl**.
- 3) Permitir iniciar sesiones *ssh* desde el sistema anfitrión y como usuario *root* autenticándose con el método de clave pública y privada. Esto le permitirá iniciar sesiones y ejecutar órdenes en la máquina virtual sin necesidad de introducir la palabra de paso del usuario *root*.

Paso 4: configuración del sistema anfitrión para poder realizar conexiones TCP/IP a la máquina virtual creada utilizando el identificador FQDN de esta.

Este último paso lo hacemos ya que no disponemos de un servicio DNS que realice estas traducciones. Para ello, en el sistema anfitrión, se debe declarar el identificador FQDN de la máquina virtual creada y su IP en el archivo `/etc/hosts`. Esto permitirá que podamos realizar, desde el sistema anfitrión, conexiones TCP/IP con la máquina virtual utilizando su identificador FQDN. Para ello, en el sistema anfitrión, debe añadir al final del archivo `/etc/hosts`, la siguiente línea:

```
IP mvp1.vpd.com
```

Siendo IP la dirección IP de la máquina `mvp1.vpd.com`. Al final, si la dirección IP de la máquina `mvp1.vpd.com` fuera `192.168.122.104`, entonces el contenido del archivo `/etc/hosts` debería ser análogo a:

```
# Loopback entries; do not change.  
  
# For historical reasons, localhost precedes  
localhost.localdomain:
```

```
127.0.0.1    localhost localhost.localdomain localhost4
localhost4.localhostdomain4

::1         localhost localhost.localdomain localhost6
localhost6.localhostdomain6

# See hosts(5) for proper format and other examples:
# 192.168.1.10 foo.mydomain.org foo
# 192.168.1.13 bar.mydomain.org bar
192.168.122.104 mvp1.vpd.com
```

Algunas consideraciones finales

- 1) Es muy conveniente que al concluir la instalación de la plataforma KVM reinicie el sistema antes de usar la plataforma de virtualización. También es muy conveniente asegurarse que los módulos del núcleo que dan soporte a la plataforma de virtualización se han cargado durante el arranque del sistema.
- 2) Con la utilidad *virt-manager* se pueden consultar las características hardware de la máquina virtual. Es conveniente que compruebe que las características mostradas se corresponden con las indicadas en la ficha.
- 3) Observar que se puede modificar el hardware, modificando las características del hardware que tiene o añadiendo hardware virtual.
- 4) Averiguar a qué red está conectada la máquina virtual y comprobar la conectividad entre la máquina virtual y el anfitrión y entre la máquina virtual y el exterior.

4 Verificación

Cuando finalice las tareas, se realizarán las siguientes comprobaciones:

- ☐ Verificación de que el/la estudiante ha completado la encuesta “Datos PC”.
- ☐ Verificación de que *SELinux* se está ejecutando en modo “enforcing”.
- ☐ Verificación de que se han instalado todos los paquetes que forman parte del grupo **Virtualization**.
- ☐ Comprobación de que el servicio **virtnetworkd** se está ejecutando y está configurado para que se inicie con el arranque del sistema.
- ☐ Comprobación de que están cargados los módulos del núcleo **kvm** y **kvm_intel**.
- ☐ Verificación de las características básicas de configuración de la máquina virtual: nombre de la máquina, procesador, RAM, disco, interfaz de red, ...
- ☐ Verificación de la instalación del SO en la MV con nombre **mvp1.vpd.com**: método de instalación elegido y funcionamiento adecuado de la MV, comprobando su conectividad con el host anfitrión y salida al exterior (para ello utilice la orden **ping**).
- ☐ Verificación de que el agente **qemu-guest-agent** está instalado y habilitado para su ejecución durante el arranque del SO en la máquina virtual.

- ☐ Comprobación de que es posible abrir una sesión como usuario `root` en la máquina virtual vía SSH desde el sistema anfitrión, no requiriendo para ello la introducción de la palabra de paso. En la orden `ssh` debe poder emplearse el identificador FQDN de la máquina virtual.

5 Entrega

Para las tareas 1 y 2 se deberá cumplimentar la encuesta “Datos PC” que se encuentra en el curso *moodle* de la asignatura.

Para el resto de las tareas, de la 3 a la 10, se debe entregar un informe que contenga la explicación de cómo se han realizado todos los pasos y el conjunto de órdenes concretas que ha necesitado ejecutar para la realización de cada una de ellas. Para elaborar este informe, siga las especificaciones contenidas en la “**Guía para la elaboración de los informes de las prácticas obligatorios y rúbrica de evaluación**” que se encuentra disponible en el curso *moodle* de la asignatura.

6 Rúbrica de evaluación

Los/as estudiantes deberán validar el trabajo realizado en esta práctica ante su profesor/a de prácticas, preferiblemente durante el horario de prácticas de laboratorio. Será responsabilidad de los/as estudiantes concertar una cita con el/la profesor/a correspondiente si desean realizar la defensa en otro momento:

- Un estudiante obtendrá una calificación de 8 en esta práctica si supera la validación de la misma y además entrega el informe que contiene las evidencias del trabajo realizado dentro del plazo estipulado.
- Un estudiante obtendrá una calificación de 10 en esta práctica si el supera la validación de la misma, entrega el informe que contiene las evidencias del trabajo realizado dentro del plazo estipulado y además, en dicho informe se explican los parámetros y el sentido de cada una de las órdenes que se emplean para desarrollar cada una de las tareas.

Por otro lado, si un estudiante concluye las tareas de esta práctica obligatoria de forma satisfactoria en las sesiones oficialmente programadas para su desarrollo por el equipo docente (3 sesiones en el caso de esta práctica), el estudiante obtendrá una calificación acumulable en el ítem de “Participación activa” que se explica en el proyecto docente de 0,143. La correcta culminación de las tareas será verificada por el profesor responsable en la última sesión de la práctica, no incluyéndose la realización del informe de la práctica.

7 Consideraciones adicionales

No realice NUNCA, es decir, ni en esta práctica ni en las venideras, una actualización del sistema operativo instalado en el sistema anfitrión. Esto es, nunca ejecute la orden `dnf` para realizar una actualización del sistema (`update` o `upgrade`). El motivo de esta norma es que en el PC hay también instalados otros

sistemas operativos utilizados en otras asignaturas. Una actualización (`update` o `upgrade`) del sistema anfitrión podría provocar que el arranque de los distintos sistemas instalados no se ejecutase correctamente.

Fuentes de información

- [1] Herrmann J, Zimmerman Y, Parker D, Novich L, East J, Radvan S. *Red Hat Enterprise Linux 7. Virtualization Getting Started Guide. Introduction to virtualization technologies available with RHEL*, Red Hat; 2019. Disponible en: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/virtualization_getting_started_guide/index [accedido el 20/01/2025]
- [2] Herrmann J, Zimmerman Y, Novich L, Parker D, Radvan S, Richardson T. *Red Hat Enterprise Linux 7. Virtualization Deployment and Administration Guide. Installing, configuring, and managing virtual machines on a RHEL physical machine*, Red Hat; 2019. Disponible en: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/virtualization_deployment_and_administration_guide/index [accedido el 20/01/2025]
- [3] Peter Boy, Jan Kupařinen, Peter W. Smith. Fedora Server Documentation.. Disponible en: <https://docs.fedoraproject.org/en-US/fedora-server> [accedido el 20/01/2025]
- [4] Christie M, Prabhui S, Evers R, Howels D, Lehman D, Moyer J et al. *Red Hat Enterprise Linux 7. Storage Administration Guide. Deploying and configuring single-node storage in RHEL 7*. Red Hat; 2019. Disponible en: https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/system_administrators_guide/index [accedido el 20/01/2025]