

Cuando se busca la solución de virtualización más potente y nativa para el ecosistema Linux, la respuesta es un conjunto de herramientas que trabajan en perfecta sintonía. A diferencia de un programa único como VirtualBox, aquí utilizamos un *stack* o conjunto de tecnologías de nivel profesional que se integran directamente con el sistema operativo. Para entender cómo funciona, es clave conocer a sus componentes. Piensa en ellos como las piezas de un motor de alto rendimiento:

- `[[QEMU]]` : Es el **motor principal**. El software que ejecuta la máquina virtual y simula sus componentes de hardware (disco, red, gráficos).
 - `[[KVM]]` : Es el **acelerador por hardware**. Una tecnología dentro del propio kernel de Linux que permite a `[[QEMU]]` acceder directamente al procesador del anfitrión, otorgando una velocidad casi idéntica a la de un sistema real.
 - `[[Libvirt]]` : Es el **panel de control estandarizado**. Un servicio que funciona por debajo y nos permite administrar de forma segura y consistente las máquinas virtuales, sus redes y su almacenamiento.
 - `[[Virt-Manager]]` : Es nuestra **interfaz gráfica**. La aplicación de escritorio que usamos para interactuar de forma sencilla con `[[Libvirt]]` , permitiéndonos crear, configurar e iniciar nuestras máquinas virtuales con unos pocos clics.
- Esta guía te llevará a través del proceso de instalar y configurar este potente stack de virtualización en las principales distribuciones de Linux.

Ventajas Principales

- **Rendimiento Casi Nativo:** Gracias a la profunda integración de `[[KVM]]` con el kernel de Linux, las máquinas virtuales se ejecutan con una sobrecarga mínima, ofreciendo un rendimiento muy superior al de otras soluciones que no son nativas del sistema.
- **Integración Profunda con el Sistema:** No es una aplicación de terceros; es una característica fundamental de Linux. Esto se traduce en una mayor estabilidad, mejor gestión de recursos y una seguridad robusta que puede aprovechar mecanismos del sistema como SELinux o AppArmor.
- **Flexibilidad y Escalabilidad Profesional:** Aunque lo manejaremos con la amigable interfaz de `[[Virt-Manager]]` , todo el sistema puede ser automatizado y controlado mediante scripts. Es la misma tecnología que se usa para construir gigantescas infraestructuras en la nube (como OpenStack).
- **Estándar de la Industria y Código Abierto:** Aprender a usar este stack te proporciona habilidades directamente aplicables en el mundo profesional de la administración de sistemas. Al ser 100% de código abierto, es completamente gratuito y se beneficia de una enorme comunidad global de desarrolladores.

Instalación y configuración

A continuación, encontraras los pasos para las tres principales familias de distribuciones.

Arch Linux y Derivadas (Manjaro, EndeavourOS, Cachi OS)

Arch Linux y sus derivadas se caracterizan por ofrecer paquetes muy actualizados y un control total sobre el sistema.

1. **Instalar los paquetes necesarios:**

```
sudo pacman -S qemu virt-manager virt-viewer dnsmasq bridge-utils libvirt edk2-ovmf swtpm
```

- `qemu` : El emulador y virtualizador que actuará como el motor de nuestra máquina virtual.
- `virt-manager` : La interfaz gráfica de escritorio para gestionar las máquinas virtuales.
- `virt-viewer` : El visor gráfico que nos permite ver e interactuar con el escritorio de la máquina virtual.
- `dnsmasq` y `bridge-utils` : Herramientas necesarias para que `libvirt` cree y gestione las redes virtuales, permitiendo que nuestras VMs tengan acceso a internet.
- `libvirt` : El servicio (daemon) que se ejecuta en segundo plano y gestiona todo el ciclo de vida de las máquinas virtuales.
- `edk2-ovmf` : **(Firmware de Arranque)**. Paquete crucial que proporciona el firmware `[[UEFI]]` para nuestras máquinas virtuales, permitiéndoles arrancar como lo hacen los ordenadores modernos. Es el equivalente a la BIOS/UEFI de un PC físico.
- `swtpm` : **(Módulo de Seguridad)**. Proporciona un `[[TPM]]` (Módulo de Plataforma Confiable) virtual. Este componente es opcional para Linux, pero **indispensable si planeas instalar Windows 11**, ya que es uno de sus requisitos de seguridad.

Selecciona qemu-full

```
➔ ~ sudo pacman -S qemu virt-manager virt-viewer dnsmasq bridge-utils libvirt edk2-ovmf swtpm
[sudo] contraseña para aether:
:: Existen 3 proveedores disponibles para qemu:
:: Repositorio extra
   1) qemu-base  2) qemu-desktop  3) qemu-full

Introduzca un número (por omisión=1): 3
```

2. **Iniciar y habilitar el servicio de Libvirt:** Este comando asegura que el servicio de virtualización se inicie automáticamente cada vez que enciendes tu ordenador.

```
sudo systemctl enable --now libvirtd.service
```

3. **Añadir tu usuario al grupo libvirt :** Esto es un paso de seguridad y comodidad. Te permite crear y gestionar máquinas virtuales como tu usuario normal, sin necesidad de usar `sudo` para cada acción.

Bash

```
sudo usermod -aG libvirt $(whoami)
```

¡Importante! Después de ejecutar este comando, necesitas **cerrar sesión y volver a iniciarla** (o reiniciar el PC) para que el cambio de grupo surta efecto.

Debian y Derivadas (Ubuntu, Linux Mint, Pop!_OS)

Esta familia es conocida por su estabilidad y su vasta selección de software. Los paquetes aquí suelen agrupar funcionalidades de manera muy conveniente.

1. **Instalar los paquetes necesarios:** Este comando instalará el stack completo, incluyendo el motor, la interfaz, el soporte para UEFI y para el TPM virtual.

```
sudo apt update
sudo apt install qemu-kvm libvirt-daemon-system virt-manager bridge-utils ovmf swtpm
```

- `qemu-kvm` : El paquete principal que proporciona tanto el motor QEMU como la integración con el acelerador KVM del kernel.
 - `libvirt-daemon-system` : El servicio de `libvirt` que se ejecuta en segundo plano para gestionar todo el entorno de virtualización.
 - `virt-manager` : La interfaz gráfica de escritorio que usaremos para controlar nuestras máquinas virtuales.
 - `bridge-utils` : Herramientas para la creación de redes virtuales complejas (bridging).
 - `ovmf` : **(Firmware de Arranque)**. Es el equivalente a `edk2-ovmf` en Arch. Proporciona el firmware `[[UEFI]]` para nuestras VMs.
 - `swtpm` : **(Módulo de Seguridad)**. Proporciona el `[[TPM]]` virtual, necesario para sistemas operativos como Windows 11.
2. **Verificar el servicio de Libvirt:** En sistemas basados en Debian, el servicio generalmente se inicia y habilita por sí solo tras la instalación. Puedes confirmarlo con:

```
sudo systemctl status libvirtd.service
```

Si no está activo, puedes iniciarlo con `sudo systemctl enable --now libvirtd.service` .

3. **Añadir tu usuario a los grupos necesarios:** Esto te concede los permisos para gestionar máquinas virtuales sin usar `sudo` .

```
sudo usermod -aG libvirt $(whoami)
sudo usermod -aG kvm $(whoami)
```

¡Importante! Debes **cerrar sesión y volver a iniciarla** (o reiniciar el PC) para que los permisos se apliquen correctamente.

Red Hat y Derivadas (Fedora, CentOS Stream, Rocky Linux)

Esta familia de distribuciones, especialmente Fedora, es líder en la integración de tecnologías de virtualización y ofrece un método de instalación muy simplificado.

1. **Instalar el grupo de virtualización:** En lugar de instalar paquetes uno por uno, podemos usar el grupo `@virtualization`, que está diseñado para configurar un anfitrión de virtualización completo con todo lo necesario.

```
sudo dnf install @virtualization
```

Este único comando instalará un conjunto completo de herramientas, que incluye:

- **Motor de virtualización:** `qemu-kvm`.
- **Servicio de gestión:** `libvirt-daemon`.
- **Interfaz gráfica:** `virt-manager` y su visor `virt-viewer`.
- **Firmware de Arranque:** El soporte para `[[UEFI]]` (`edk2-ovmf`).
- **Módulo de Seguridad:** El soporte para `[[TPM]]` virtual (`swtpm`).
- **Herramientas de red:** `bridge-utils` y todo lo necesario para la red por defecto.
- Y muchas otras dependencias para una experiencia robusta.

2. **Iniciar y habilitar el servicio de Libvirt:**

```
sudo systemctl enable --now libvirtd.service
```

3. **Añadir tu usuario al grupo `libvirt`:**

```
sudo usermod -aG libvirt $(whoami)
```

¡Importante! No olvides **cerrar sesión y volver a iniciarla** para que los cambios de permisos tengan efecto.