

PROXMOX VE · SESIÓN 2 — USO BÁSICO DE PROXMOX VE

Uso básico de Proxmox VE

Sesión 2 · Uso básico de Proxmox VE

✉ José Domingo Muñoz

🏫 IES Gonzalo Nazareno · Dos Hermanas

🌐 josedom24.github.io/curso_proxmox_2026

📄 github.com/josedom24/curso_proxmox_2026

01

DEMO 1: Gestión de máquinas virtuales Linux

Creación, configuración y administración

Almacenamiento y redes disponibles en Proxmox VE

ALMACENAMIENTO

`local` (Directory)

- Backups
- Imágenes ISO
- Plantillas de CT

`local-lvm` (LVM-Thin) (3,7 TB)

- Discos de máquinas virtuales
- Sistemas de ficheros de CTs
- Crecimiento automático

REDES VIRTUALES

`vbr0` (Linux Bridge)

- Bridge virtual para VMs y CTs
- DHCP automático (172.22.x.x)
- Accesibles desde la red local
- Por defecto: todas las VMs/CTs aquí
- Tenemos más bridge para las prácticas

Creación de una MV Linux en Proxmox (1)

- Usamos el botón **Crear VM**.
- **Nombre** obligatorio;
- Si estamos usando **Conjuntos de recursos** para guardar la MV indicamos el nombre.
- Elegimos la **ISO** del almacenamiento local.
- Seleccionamos el **tipo de SO y versión**.

- Tarjeta gráfica, tipo de máquina, BIOS y controlador VirtIO SCSI → **valores por defecto**.
- Escogemos los **discos**; por defecto se añade uno. **Podemos añadir más**.
- Disco en **local-lvm**; indicar el **tamaño**.

Creación de una MV Linux en Proxmox (2)

- **Zócalo (Socket)** — ranura física en la placa base.
- **Núcleo (Core)** — unidad de procesamiento dentro del socket.
- Elegimos cuántos **Sockets y Cores** tendrá la MV.

Elige **host** para virtualización anidada (la MV hereda las características de CPU del host).

- Indicamos la **cantidad de memoria**.
- **Red**: conectada a **vmbr0** → IP por DHCP.
- Modelo de tarjeta **VirtIO** y MAC predeterminados.

Gestión de máquinas virtuales

Acciones disponibles mediante **botón derecho** sobre la MV:

- **Iniciar** — inicia la ejecución.
- **Pause** — pausa; reanudar con **Resume**.
- **Hibernate** — guarda estado en memoria.
- **Cierre ordenado** — apagado limpio.

- **Parar** — termina inmediatamente.
- **Clonar** — duplica la máquina.
- **Convertir a plantilla** — crea una plantilla.
- **Consola** — abre terminal de la MV.

Gestión de máquinas virtuales

Secciones del **panel lateral**:

- **Resumen** — estado y monitorización.
- **Consola** — terminal de la máquina.
- **Hardware** — ver y modificar el hardware.
- **Opciones** — configurar parámetros.

- **Historial de Tareas** — registro de acciones.
- **Copia de seguridad** — gestión de backups.
- **Snapshot** — crear/restaurar instantáneas.

Monitor: la consola del hipervisor QEMU

⚠ Monitor NO es la terminal del SO invitado — es la interfaz de control del hipervisor sobre esa VM. Confusión típica del alumnado.

Proxmox lanza un proceso QEMU/KVM por cada VM. Ese proceso expone un canal de control llamado **QEMU Monitor**: hablas con el hipervisor *sobre* la VM, no con el SO que corre dentro.

Información del sistema emulado

```
info status      → estado (running, paused...)
info block       → discos y archivos del host
info network     → tarjetas de red virtuales
info cpus / info kvm / info memory-devices
```

Acciones en caliente


```
sendkey ctrl-alt-delete → útil en Windows
system_powerdown        → botón ACPI apagado
stop / cont              → pausar / reanudar CPU
change                   → cambiar medio CD-ROM
screendump /tmp/cap.ppm → captura de pantalla
```

Eliminar una máquina virtual

 La máquina debe estar **parada** antes de poder eliminarla.

- Escoger la opción **Eliminar** del botón **Más**.
- Se pedirá el **identificador** de la MV para confirmar.

Opciones de máquinas virtuales

 Algunos cambios requieren **reinicio** para aplicarse.

- **Nombre, Tipo de OS, Orden de arranque.**

Hardware de la máquina virtual

- Ver y modificar RAM, CPU, BIOS, Display y dispositivos conectados.
- Podemos **añadir nuevos dispositivos** y eliminarlos.

⚠ El cambio de hardware requiere **reinicio** de la máquina.

Qemu-guest-agent

Demonio en la MV que permite comunicación con Proxmox: muestra la **IP**, facilita apagados ordenados y snapshots consistentes. Activar la opción en la configuración de la MV.

```
apt install qemu-guest-agent
```

RECOMENDADO EN TODAS LAS MV

02

DEMO 2: Gestión de máquinas virtuales Windows

Particularidades y configuración específica

Windows en Proxmox

Los pasos son similares a los de una MV Linux, pero con una diferencia clave:

¿Qué cambia?

- Disco duro e interfaz de red de tipo **VirtIO** → mayor rendimiento.
- Windows **no incluye** los drivers para dispositivos VirtIO.

Solución

Añadir un segundo CDROM con los **drivers VirtIO** de Proxmox antes de iniciar la instalación.

 Debes subir al **almacenamiento local** la ISO de los drivers Virtio ([Descarga](#)).

 ([Windows VirtIO Drivers](#)).

Creación de una MV Windows en Proxmox

- SO: **Microsoft Windows** + versión correspondiente.
- Tipo de disco **Bus/device** → **VirtIO Block**.

- **Añadir CDROM con drivers VirtIO**
- Máquina → **Hardware** → **Agregar** → **Dispositivo CD/DVD**.
- **Configurar orden de arranque**
- **Opciones** → **Orden de arranque**: la ISO de Windows debe estar por encima de la ISO de drivers VirtIO.

i Una vez configurado el orden de arranque, ya podemos **iniciar la máquina** .

Comienzo de la instalación

Durante la instalación, Windows no detecta el disco duro porque no tiene los drivers VirtIO.

1. En la pantalla de selección de disco → **Cargar contr.** → **Examinar.**
2. Del CDROM de drivers seleccionamos la carpeta `amd64` de nuestra **versión de Windows.**
3. Con el driver cargado, Windows detecta el disco y podemos continuar.

 Sin este paso **no aparecerá ningún disco** disponible para la instalación.

Configuración de red

⚠ Sin drivers VirtIO, Windows **no tiene conexión de red** tras la instalación.

Instalar drivers desde el **Administrador de dispositivos**:

1. Buscar **Controladora Ethernet** → **Actualizar controlador**.
2. Seleccionar la carpeta del CDRROM de drivers VirtIO:

```
NetKVM\<<versión_windows>\amd64
```

Qemu-guest-agent

Demonio que permite comunicación con Proxmox: muestra la **IP**, facilita apagados ordenados y snapshots consistentes. Activar la opción en la configuración de la MV.

En la MV Windows:

1. **Administrador de Dispositivos.**
2. Buscar **PCI Simple Communications Controller.**
3. **Actualizar controlador** → seleccionar del CDROM de drivers VirtIO.

RECOMENDADO EN TODAS LAS MV

03

DEMO 3: Gestión de contenedores LXC

Contenedores ligeros y su administración

Contenedores en Proxmox

¿Qué es un contenedor?

Conjunto de procesos que se ejecutan en el host con su propio **sistema de ficheros** y **configuración de red**. Se puede considerar una **mini máquina virtual**.

- Comparten el **kernel del host** → más ligeros y rápidos.
- Todos los contenedores son **Linux**.

Ejemplos: Docker, LXC, ...

En Proxmox trabajamos con contenedores **LXC**.

Mayor eficiencia que las MV a cambio de menor aislamiento respecto al host.

Crear un contenedor

Desde el botón **Crear CT** lanzamos el asistente de configuración.

Campos obligatorios

- **Nombre del Host** — identificador del contenedor.
- **Conjunto de recursos** (si lo estamos utilizando).
- **Contraseña** del usuario **root**.

Opcional

- **Clave SSH pública** para acceso por SSH sin contraseña.

Elegimos el sistema operativo (Plantilla)

¿Qué es una plantilla?

Imagen de SO preconfigurada que sirve de base para el contenedor. Se descarga del repositorio de Proxmox y se almacena en el **almacenamiento local**.

Configuración del contenedor (similar a las MV):

- **Disco** — tamaño del almacenamiento.
- **CPU** — número de cores.
- **Memoria** — RAM asignada.
- **Red** — interfaz y configuración de red.

i La gestión y eliminación de contenedores es **similar a la de las MV** .

Inicio del contenedor

⚠ Al crear el contenedor, el único usuario disponible es **root** .

- Iniciamos el contenedor desde el panel de control.
- Accedemos a él mediante la **Consola** de Proxmox o por **SSH**.

04

Gestión desde la línea de comandos

qm, pct y pvesh

Tres herramientas CLI

HERRAMIENTA	QUÉ GESTIONA	CUÁNDO USARLA
<code>qm</code>	Máquinas virtuales (KVM)	Administración y scripting de MVs
<code>pct</code>	Contenedores LXC	Administración y scripting de CTs
<code>pvesh</code>	Toda la API REST de Proxmox	Automatización avanzada, cualquier recurso

- `qm` y `pct` operan sobre el nodo local.
- `pvesh` accede a la API REST del clúster y puede gestionar recursos de cualquier nodo.

qm — Gestión de MVs desde el terminal

LISTAR Y CONSULTAR

```
qm list           # todas las MVs
qm status <vmid>  # estado de una MV
qm config <vmid>  # configuración completa
```

CREAR

```
qm create 200 \  
  --name debian-test \  
  --memory 2048 --cores 2 \  
  --net0 virtio,bridge=vmbr0 \  
  --scsi0 local-lvm:20 \  
  --cdrom local:iso/debian-13.1.0-amd64-netinst.iso
```

CICLO DE VIDA

```
qm start <vmid>    # iniciar
qm shutdown <vmid> # apagado ordenado
qm stop <vmid>     # parar (forzado)
qm suspend <vmid>  # pausar
qm resume <vmid>   # reanudar
qm reboot <vmid>   # reiniciar
```

ELIMINAR

```
qm stop <vmid>      # la MV debe estar parada
qm destroy <vmid> --purge # elimina MV, discos y backups
```

pct — Gestión de contenedores LXC desde el terminal

LISTAR Y CONSULTAR

```
pct list           # todos los contenedores
pct status <ctid> # estado de un CT
pct config <ctid> # configuración completa
```

CREAR

```
pct create 300 \
  local:vztmpl/debian-12-standard_12.7-1_amd64.tar.zst \
  --hostname ct-debian \
  --memory 512 --cores 1 \
  --net0 name=eth0,bridge=vbr0,ip=dhcp \
  --rootfs local-lvm:8 \
  --password secreto
```

CICLO DE VIDA

```
pct start <ctid>      # iniciar
pct shutdown <ctid>  # apagado ordenado
pct stop <ctid>       # parar (forzado)
pct suspend <ctid>   # pausar
pct resume <ctid>    # reanudar
pct reboot <ctid>    # reiniciar
pct enter <ctid>     # abre shell dentro del CT
```

ELIMINAR

```
pct stop <ctid>      # el CT debe estar parado
pct destroy <ctid> --purge # elimina CT, discos y backup
```

pvesh — La API REST desde el terminal

⚠ El nombre del nodo es `proxmox1` . Puedes verificarlo con: `pvesh get /nodes`

LISTAR RECURSOS

```
pvesh get /nodes/proxmox1/qemu # listar MVs
pvesh get /nodes/proxmox1/lxc # listar contenedores

# Estado de una MV
pvesh get /nodes/proxmox1/qemu/200/status/current
```

ELIMINAR

```
pvesh delete /nodes/proxmox1/qemu/200 # MV
pvesh delete /nodes/proxmox1/lxc/300 # contenedor
```

CICLO DE VIDA — MVS

```
pvesh create /nodes/proxmox1/qemu/200/status/start
pvesh create /nodes/proxmox1/qemu/200/status/shutdown
pvesh create /nodes/proxmox1/qemu/200/status/stop
pvesh create /nodes/proxmox1/qemu/200/status/suspend
pvesh create /nodes/proxmox1/qemu/200/status/resume
```

CICLO DE VIDA — CONTENEDORES

```
pvesh create /nodes/proxmox1/lxc/300/status/start
pvesh create /nodes/proxmox1/lxc/300/status/shutdown
pvesh create /nodes/proxmox1/lxc/300/status/stop
```

¿Cuándo usar cada herramienta?

QM

- Tareas cotidianas sobre **MVs** desde el terminal
- Scripts de automatización con MVs
- Equivalente CLI de la interfaz web para MVs

PCT

- Tareas cotidianas sobre **contenedores** desde el terminal
- Scripts de automatización con CTs
- Equivalente CLI de la interfaz web para CTs

PVESH

- Recursos sin comando específico en `qm / pct`
- Integración con la **API REST** de Proxmox
- Explorar la API interactivamente

i En el día a día **qm y pct son suficientes** . Usa `pvesh` cuando necesites algo que no cubran directamente.

Recursos

- [Curso de introducción a Proxmox VE \(CEP Castilleja de la Cuesta\)](#)
 - Capítulo 3: Creación de máquinas virtuales
 - Capítulo 7: Trabajando con Linux Containers
- Prácticas con alumnos:
 - [Práctica 1: Instalación de una máquina Linux en Proxmox](#)
 - [Práctica 2: Instalación de una máquina Windows en Proxmox](#)
 - [Práctica 3: Trabajo con claves SSH](#)
 - [Práctica 6: Instalación de contenedores en Proxmox](#)
- [Trabajando con claves ssh](#)
- [Escritorio remoto con Debian GNOME](#)

PROXMOX VE · SESIÓN 2 — USO BÁSICO DE PROXMOX VE

¡Gracias!

Sesión 3 → Almacenamiento y redes en Proxmox VE

 José Domingo Muñoz

 IES Gonzalo Nazareno · Dos Hermanas

 https://josedom24.github.io/curso_proxmox_2026