

CURSO DE VIRTUALIZACIÓN CON PROXMOX VE

José Luis Brito
Octubre de 2022

Versión: 0.1



Este obra está bajo una [licencia de Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-sa/4.0/).

Indice

Introducción a la Virtualización.....	3
Requisitos de Hardware de Proxmox1.....	4
Descarga de la imagen ISO de instalación.....	5
Preparación del medio de instalación.....	7
Repositorios de Proxmox.....	10
Instalación de Proxmox en un ISP.....	12
Tipos de virtualización con Proxmox VE.....	14
Contenedores, OpenVZ y LXC.....	15
Creación de Contenedores.....	16
Servidor NFS con Debian GNU/Linux.....	24
Agregar un Storage NFS a Proxmox.....	27
Carga de imágenes ISO hacia Proxmox.....	29
Creación de una Maquina Virtual.....	31
Creación de Templates en Proxmox.....	36
Creación de cuentas de Usuario.....	38
Grupos y asignación de Permisos a usuarios.....	41
Automatización de Backups.....	45
Copia de una VM desde una unidad usb.....	46
Protección de Proxmox mediante fail2ban.....	47

Introducción a la Virtualización

La virtualización de sistemas nos permite crear entornos con hardware simulado, en donde poder ejecutar sistemas operativos mediante la utilización de dichos recursos virtuales.

Podemos diferenciar dos métodos para virtualización, **hipervisores de tipo 1 e hipervisores de tipo 2.**

En los hipervisores de tipo 1 (también conocidos como “hipervisores bare metal” o “hipervisores nativos”) existe un sistema operativo que se encarga de soportar n copias del hardware actual, conocidas como “maquinas virtuales” (en adelante VM, de “Virtual Machine”), además de controlar la asignación de los recursos disponibles para dichas VM.

En los hipervisores de tipo 2, el método utilizado es totalmente distinto; se utiliza solo un programa de usuario que se ejecuta en el sistema operativo “anfitrión”.

Por ejemplo en Windows (también en Linux) disponemos de VirtualBox (el cual se ejecuta como un programa dentro del sistema operativo, y permite crear VM’s).

Tanto en los hipervisores tipo 1 como tipo 2, llamamos “sistema operativo invitado” (en ingles “Guest OS”) a la VM que se ejecuta sobre el sistema de virtualización, y “sistema operativo anfitrión” (en ingles “Host OS”) al sistema operativo base que permite la virtualización.

A partir de la versión 2.6.20 del Kernel de Linux, disponemos de la tecnología KVM (Kernel-Based Virtual Machine) que nos permite utilizar Linux como un hipervisor de tipo 1. Para hacer uso de KVM podemos utilizar la api libvirt, que permite interactuar con las capacidades de virtualización; existen aplicativos como Virt-Manager, que permiten controlar la creación

y ejecución de VM's desde una interfaz grafica, pero accediendo localmente como usuario del sistema anfitrión (y teniendo permisos administrativos que permitan dicho acceso).

Otra opción, es hacer uso de una distribución dedicada a proveer un entorno de virtualización más profesional, también haciendo uso de KVM, en este caso disponemos de la distribución de GNU/Linux llamada **PROXMOX VE**, basada en Debian GNU/Linux; provee virtualización KVM y OpenVZ, junto a una interface web para la administración tanto del sistema anfitrión (acceso a la consola o "shell") como de las VM (y los contenedores).

Este documento se centra en mostrar una introducción a las principales funcionalidades de PROXMOX VE (backup, uso de storage NFS, creación de templates, gestión de usuarios) además de como descargar el sistema desde internet e instalar el mismo.

Requisitos de Hardware de Proxmox¹

Para servidores de producción es requerido equipamiento de calidad a nivel servidor (equipos especiales para uso en centros de computo que incluyan tecnología como fuentes de alimentación redundantes, controladoras raid y discos SAS por citar ejemplos). Las instalaciones de Proxmox VE pueden ser manejadas en forma centralizada gracias a funciones de integración en forma de Cluster. Proxmox VE puede hacer uso de almacenamiento local, almacenamiento compartido y almacenamiento distribuido (Ceph).

Hardware recomendado¹:

- Intel EMT64 or AMD64 with Intel VT/AMD-V CPU flag.
- Memory, minimum 2 GB for OS and Proxmox VE services. Plus designated memory for guests. For Ceph or ZFS additional memory is required, approximately 1 GB memory for every TB used storage.
- Fast and redundant storage, best results with SSD disks. OS storage: Hardware RAID with batteries protected write cache ("BBU") or non-RAID with ZFS and SSD cache.

- VM storage: For local storage use a hardware RAID with battery backed write cache (BBU) or non-RAID for ZFS. Neither ZFS nor Ceph are compatible with a hardware RAID controller. Shared and distributed storage is also possible.
- Redundant Gbit NICs, additional NICs depending on the preferred storage technology and cluster setup – 10 Gbit and higher is also supported.
- For PCI(e) passthrough a CPU with VT-d/AMD-d CPU flag is needed.

Hardware mínimo (evaluación de la distribución):

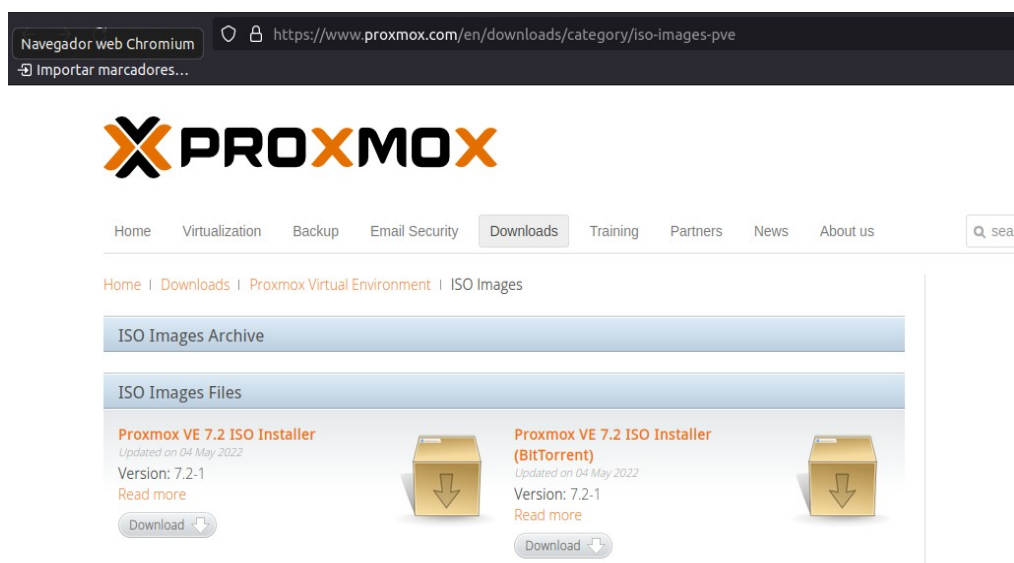
- CPU: 64bit (Intel EMT64 or AMD64)
- Intel VT/AMD-V capable CPU/Mainboard (for KVM Full Virtualization support)
- Minimum 1 GB RAM
- Hard drive
- One NIC

1. Referencia: <https://www.proxmox.com/en/proxmox-ve/requirements>

Descarga de la imagen ISO de instalación

Para obtener la imagen ISO, ir a la dirección

<https://www.proxmox.com/en/downloads/category/iso-images-pve>

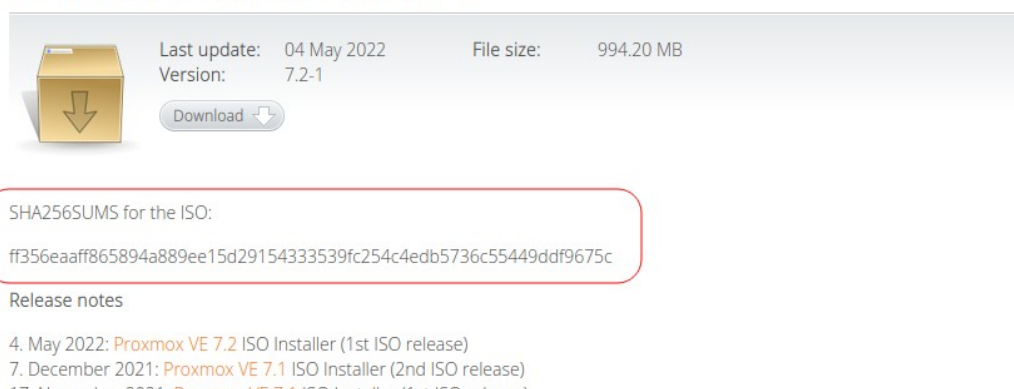


Pulsar en el botón “Download” para realizar la descarga.

Curso de Virtualización con Proxmox

Luego de descargar la imagen, procederemos a la comprobación de integridad de la misma, para lo cual crearemos un archivo llamado SHA256SUM (sin extensión) en el cual pegaremos la suma de comprobación provista en la pagina seguida de un `*nombre_del_archivo.iso`

Proxmox VE 7.2 ISO Installer



Last update: 04 May 2022
Version: 7.2-1
File size: 994.20 MB

Download

SHA256SUMS for the ISO:

ff356eaaff865894a889ee15d29154333539fc254c4edb5736c55449ddf9675c

Release notes

4. May 2022: Proxmox VE 7.2 ISO Installer (1st ISO release)
7. December 2021: Proxmox VE 7.1 ISO Installer (2nd ISO release)
17. November 2021: Proxmox VE 7.1 ISO Installer (1st ISO release)

Ejemplo archivo SHA256SUM:



Luego de crear el archivo SHA256SUM, procedemos a verificar por terminal que la suma coincida, ejecutando el comando:

```
sha256sum -c SHA256SUM
```

```
~/Escritorio/CURSO_PROXMOX/INSTALL$ sha256sum -c SHA256SUM
proxmox-ve_7.2-1.iso: La suma coincide
~/Escritorio/CURSO_PROXMOX/INSTALL$
```

Preparación del medio de instalación

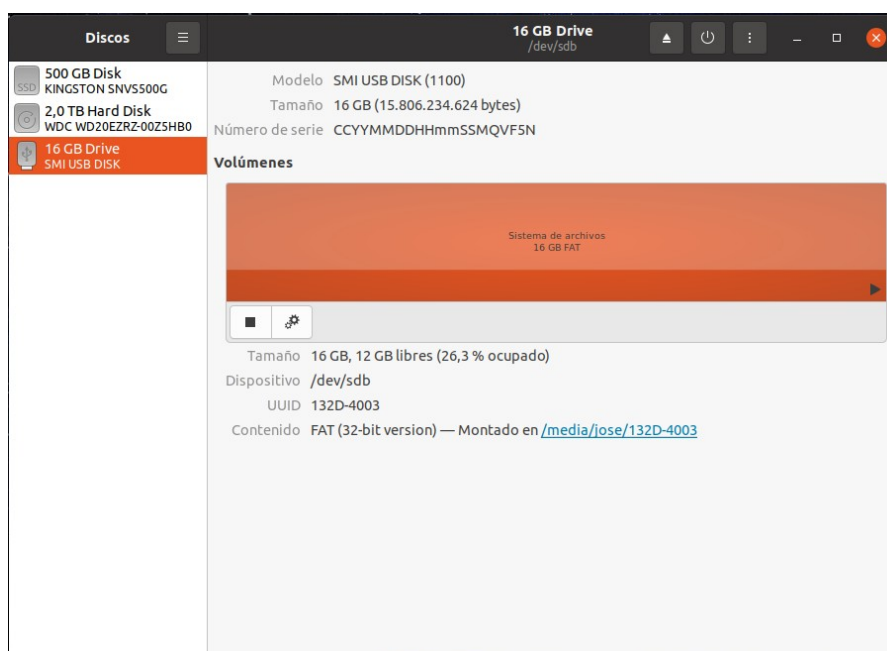
Luego de obtenida y comprobada la integridad de la imagen ISO de instalación, debemos crear un medio de instalación (unidad USB por ejemplo), para lo cual seguiremos estos pasos:

1. Verificar la tabla de particiones usando el comando `lsblk` (en este caso la unidad usb se encuentra en `devsdb`, también podemos hacer uso de la utilidad grafica “Disk”, o “Discos” del entorno Gnome)

Resultado de “`lsblk`” (terminal):

```
Disco /dev/sdb: 14,74 GiB, 15806234624 bytes, 30871552 sectores
Disk model: USB DISK
Unidades: sectores de 1 * 512 = 512 bytes
Tamaño de sector (lógico/físico): 512 bytes / 512 bytes
Tamaño de E/S (mínimo/óptimo): 512 bytes / 512 bytes
Tipo de etiqueta de disco: dos
Identificador del disco: 0x00000000
jose@dev-pc:~$
```

Resultado de “Discos” (gnome):



2. Sabiendo la unidad de destino, ejecutamos el siguiente comando:

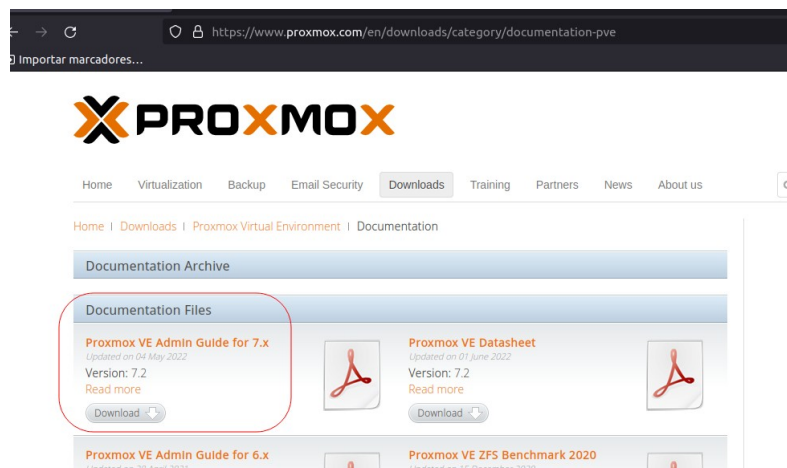
```
dd bs=1M conv=fdatasync if=./proxmox-ve_7.2-1.iso of=/dev/XYZ
```

(siendo /dev/XYZ nuestra unidad usb, prestar atención a este punto! Ejecutar este comando con permisos elevados, anteponiendo "sudo" o como root)

3. Colocar el pendrive en un puerto usb del equipo donde se instalará proxmox y comenzar la instalación seleccionando la unidad usb como dispositivo de arranque.

Luego de comenzada la instalación, continuar la misma siguiendo los pasos recomendados en la guía oficial de instalación, ubicada en: <https://pve.proxmox.com/wiki/Installation> (no esta pensado en el alcance de este documento mencionar paso a paso todo el procedimiento de instalación, pero si podemos comentar que a grandes rasgos la instalación preguntara por una clave para usuario root, parámetros de ip y hostname, distribución de teclado y mostrará una utilidad para realizar el particionado del disco duro)

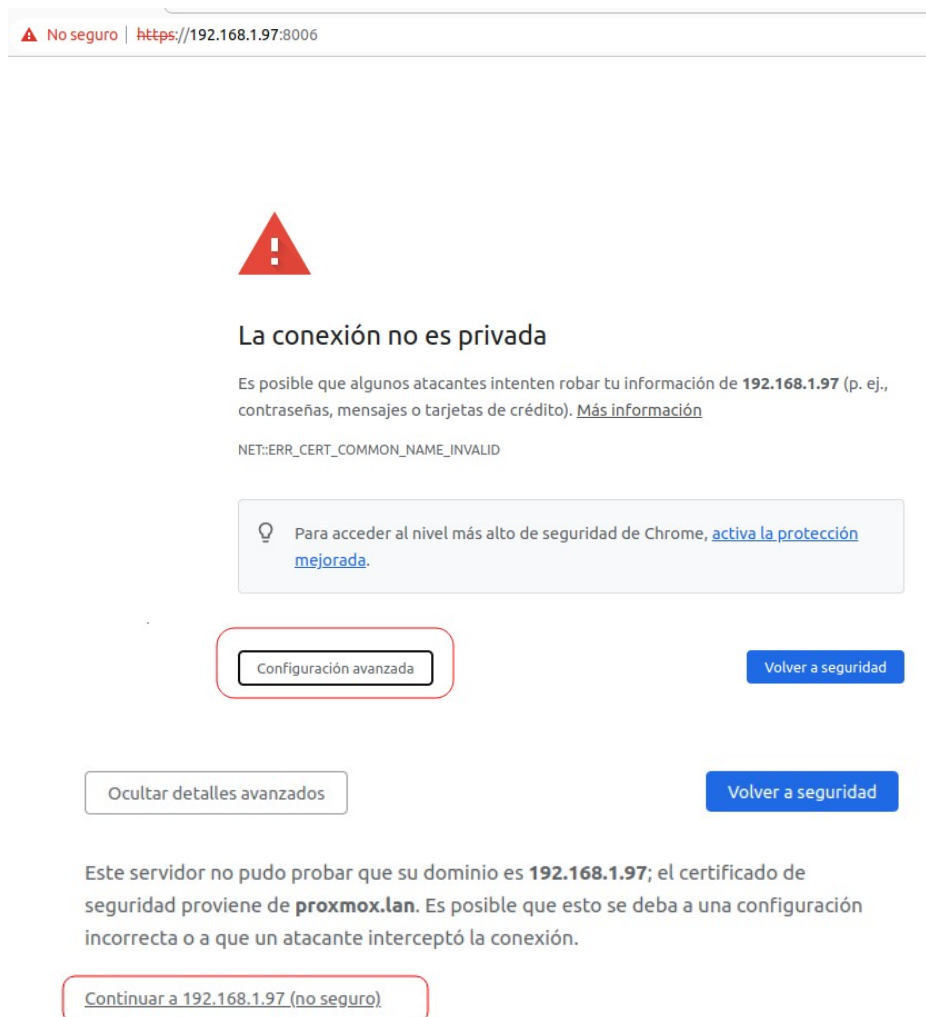
Invitamos a visitar la wiki oficial de proxmox en: https://pve.proxmox.com/wiki/Main_Page y también a realizar la descarga de la guía oficial de administración de proxmox en: <https://proxmox.com/en/downloads/category/documentation-pve>

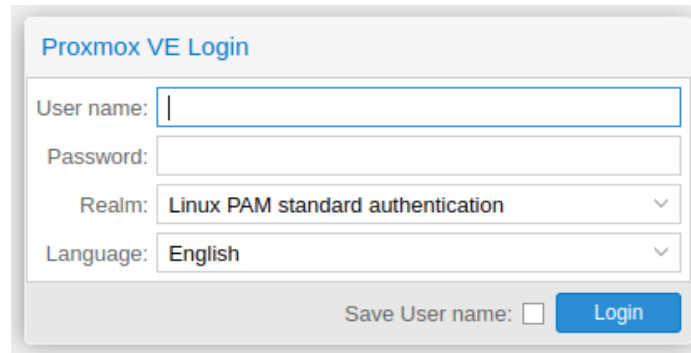


Una vez terminada la instalación del sistema, acceder a través del navegador web utilizando la ip seleccionada seguida del puerto 8006, ejemplo:

<https://192.168.1.97:8006>

Al ser un certificado auto-firmado, no será reconocido por el navegador, debemos hacer clic en “Configuración avanzada” y “Continuar a 192.168.1.97 (no seguro)”



The image shows the Proxmox VE Login interface. It has a title bar 'Proxmox VE Login'. Below it are four input fields: 'User name:' with a text box, 'Password:' with a text box, 'Realm:' with a dropdown menu showing 'Linux PAM standard authentication', and 'Language:' with a dropdown menu showing 'English'. At the bottom right, there is a checkbox labeled 'Save User name:' and a blue 'Login' button.

Una vez hayamos ingresado a la web de administración de proxmox, ingresamos con el usuario “root” y la clave que hayamos usado en el proceso de instalación.

Por defecto, el único usuario que accederá con la autenticación “Linux PAM” será el root (los usuarios que creemos desde la web administrativa deberán seleccionar la opción “Proxmox VE authentication server”)

Repositorios de Proxmox

Proxmox cuenta con repositorios test, comunitarios y empresariales; los comunitarios no requieren suscripción a diferencia de los empresariales que si la requieren (un pago anual con un costo que depende de la cantidad de sockets por servidor)

Los repositorios comunitarios no son testeados en la misma forma que los empresariales, por lo tanto desde el sitio oficial de proxmox no recomiendan su uso para entornos en producción

Curso de Virtualización con Proxmox

Para instalar los repositorios comunitarios (no recomendado para producción) se debe editar y agregar en el fichero `/etc/apt/sources.list` :

```
# PVE pve-no-subscription repository provided by proxmox.com,  
# NOT recommended for production use  
deb http://download.proxmox.com/debian/pve bullseye pve-no-  
subscription
```

Luego ejecutar

```
# apt update  
  
# apt upgrade
```

Para deshabilitar el repositorio Enterprise (por defecto habilitado) en caso de no tener una suscripción:

Renombrar el archivo `/etc/apt/sources.list.d/pve-enterprise.list` como `/etc/apt/sources.list.d/pve-enterprise.list.bkp`

Luego ejecutar

```
# apt update  
  
# apt upgrade
```

Para más información sobre los repositorios, repositorios de otras versiones, o como cambiar la información de los repositorios desde la interfaz gráfica, ir al enlace de la wiki de proxmox:

https://pve.proxmox.com/wiki/Package_Repositories

Instalación de Proxmox en un ISP

Algunos proveedores de servicios de internet (ISP) ofrecen además de servidores privados virtuales, otra opción alternativa: rentar un servidor físico (no virtualizado). Generalmente con cada servidor se ofrece una distribución de Linux, existiendo la posibilidad de optar por Debian GNU/Linux, la cual es la distribución base de Proxmox. Esto nos permite utilizar dicha infraestructura como base para un servidor de virtualización Proxmox, haciendo uso de los repositorios

Algunos proveedores de hardware dedicado:

<https://ispserver.com/dedicated-server>

<https://www.namecheap.com/hosting/dedicated-servers/>

<https://www.cherryservers.com/bare-metal-dedicated-servers>

Procedimiento para instalación de Proxmox en un ISP: en el ejemplo suponemos un escenario con Debian 11 como base ya disponible y accesible desde ssh con usuario root

1: Asignar el hostname del equipo en el archivo `/etc/hostname` (ejemplo: `miserver1`)

2: Crear una resolución del FQDN (Fully Qualified Domain Name) en el archivo `etc/hosts`: (ejemplo suponiendo la ip pública sea `200.100.100.10`: `200.100.100.10 miserver1.com`)

3: Agregar los repositorios para instalar Proxmox VE: ejecutar como root:

```
#echo "deb [arch=amd64] http://download.proxmox.com/debian/pve  
bullseye pve-no-subscription" > /etc/apt/sources.list.d/pve-  
install-repo.list
```

4: Agregamos la llave del repositorio:

```
#wget https://enterprise.proxmox.com/debian/proxmox-release-  
bullseye.gpg -O /etc/apt/trusted.gpg.d/proxmox-release-  
bullseye.gpg
```

5: Actualizamos los repositorios y si hay algún paquete disponible:

```
#apt update && apt full-upgrade
```

6: Realizamos la instalación de los paquetes necesarios:

```
#apt install proxmox-ve postfix open-iscsi
```

7: Luego de la descarga, durante el proceso de instalación se nos preguntará por un par de opciones para postfix, en dicho menú seleccionamos la opción “Internet Site”, seleccionamos “OK” para avanzar y se nos preguntará por el FQDN (en este ejemplo miserver1.com) y volvemos a seleccionar OK.

Se recomienda realizar un renicio:

```
#reboot
```

Se recomienda eliminar el kernel por defecto desde el grub:

```
#apt remove linux-image-amd64 'linux-image-5.10*'
```

```
#update-grub
```

También se recomienda la eliminación del paquete os-prober:

```
#apt remove os-prober
```

Para acceder a la nueva instalación de proxmox, escribimos en el navegador:

<https://miserver1.com:8006>

Aceptamos el certificado y ya podemos utilizar las opciones que nos provee proxmox

Tipos de virtualización con Proxmox VE

Proxmox VE nos permite realizar dos tipos de virtualización:

KVM y OpenVZ**

KVM: Maquinas virtuales completas (ya sean Linux o Windows)

OpenVZ: Contenedores (permite crear imágenes “contenidas” que comparten recursos del sistema base, no es una virtualización “completa” de todo un entorno de hardware, a diferencia de KVM)
(** A partir de Proxmox VE 4.0 se pasa de OpenVZ a LXC)

Tanto de KVM como de OpenVZ, podemos crear Templates (plantillas) los cuales se crean a partir de una VM o Contenedor (según si es KVM o OpenVZ), luego de creado un template, el mismo se puede Clonar para obtener una copia idéntica y funcional (pero el template no se puede iniciar al igual que una VM o un contenedor).

Esto nos permite levantar rápidamente un nuevo recurso (servidores por ejemplo) ya que podemos clonar una plantilla y antes de iniciarla, modificarla a nuestra necesidad (por ejemplo levantar un segundo servidor Apache con otra dirección ip, conteniendo las páginas alojadas en una nueva dirección)

Contenedores, OpenVZ y LXC

OpenVZ permite ejecutar varias instancias de sistemas operativos, o Servidores Privados Virtuales, VPS en inglés (únicamente de tipo GNU/Linux, aunque las distribuciones pueden ser variadas entre los VPS's)

Cada Entorno Virtual (EV) proporciona virtualización, aislamiento de recursos y punto de comprobación. OpenVZ es base del producto llamado "Virtuozzo" (software propietario) para virtualización, de la empresa SWsoft.

A partir de la versión 4.0, como mayor cambio se produjo la introducción de LXC en favor de OpenVZ.

LXD, abreviación de Linux Container Daemon, es una herramienta para la gestión de contenedores, fue desarrollado por Canonical (dueña también de la distribución Ubuntu).

LXD utiliza a LXC como backend por medio de la librería liblxc y los GO bindings.

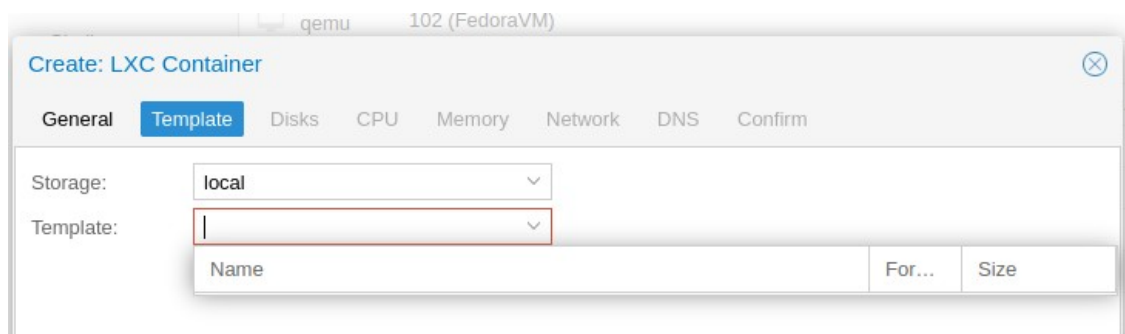
LXC (Linux Containers) son las aplicaciones virtualizadas sobre la base de un kernel linux; los contenedores comparten un sistema operativo y "aislan" los procesos del resto del sistema. Comúnmente los contenedores consumen menos recursos que la virtualización completa.

La idea de detrás de LXC es facilitar la creación de entornos que se asemejan lo mas posible a una instalación estándar de linux, sin necesidad de un kernel independiente por cada entorno.

Más información sobre LXC en la página oficial del proyecto:
<https://linuxcontainers.org/lxc/introduction/>

Creación de Contenedores

En una instalación por defecto de Proxmox VE, no disponemos de los templates necesarios para crear un contenedor a partir del procedimiento estándar, como se muestra en la imagen a continuación:



(por defecto no se muestran templates en el desplegable)

Por tanto, procederemos a descargar los mismos.

Podemos hacerlo desde el shell, y gráficamente.

Desde el shell utilizamos el comando `pveam` opciones:

pveam update: Actualiza la base de datos de Templates de los containers

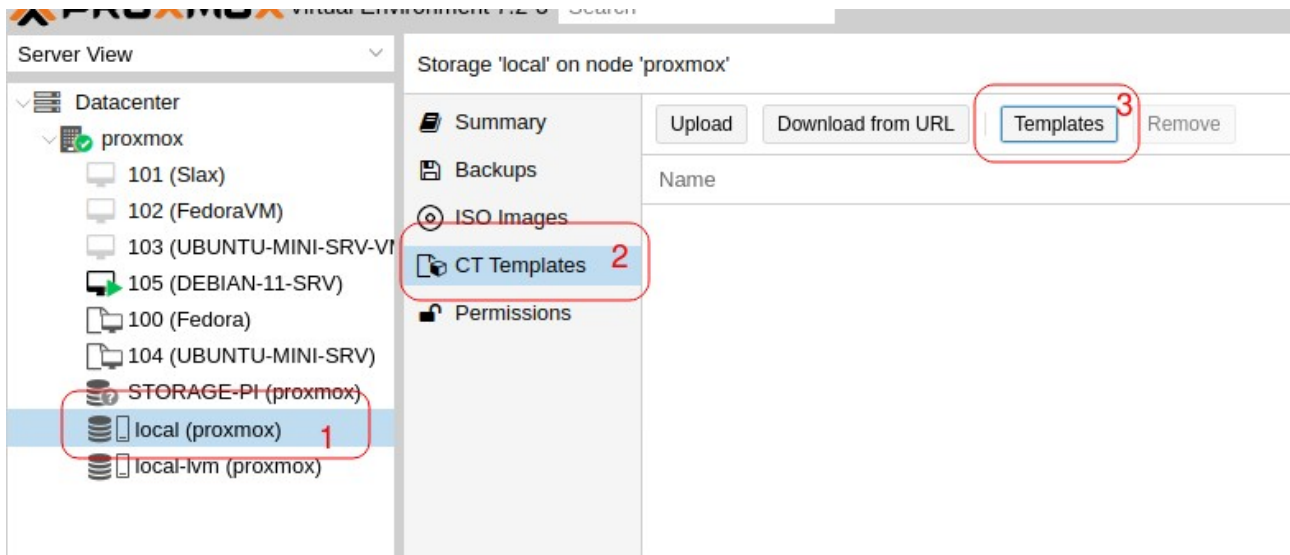
pveam list <storage>: muestra una lista de templates dentro del storage

pveam download <storage> <template>: permite la descarga de appliance templates

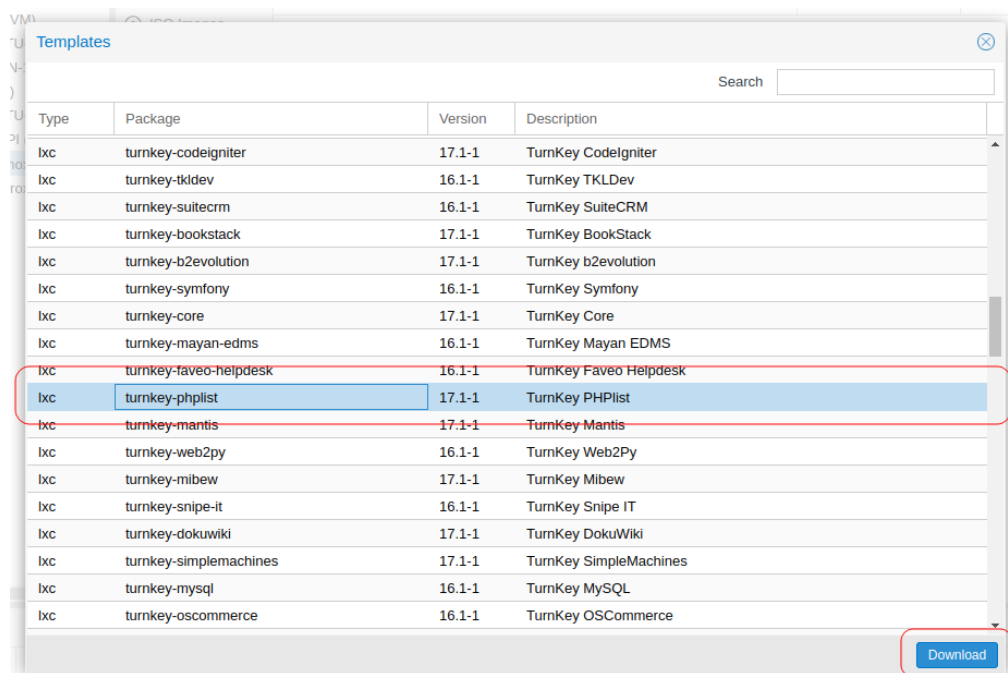
pveam available: muestra una lista de templates

Curso de Virtualización con Proxmox

Forma gráfica: Seleccionamos el almacenamiento local, luego CT Templates y templates

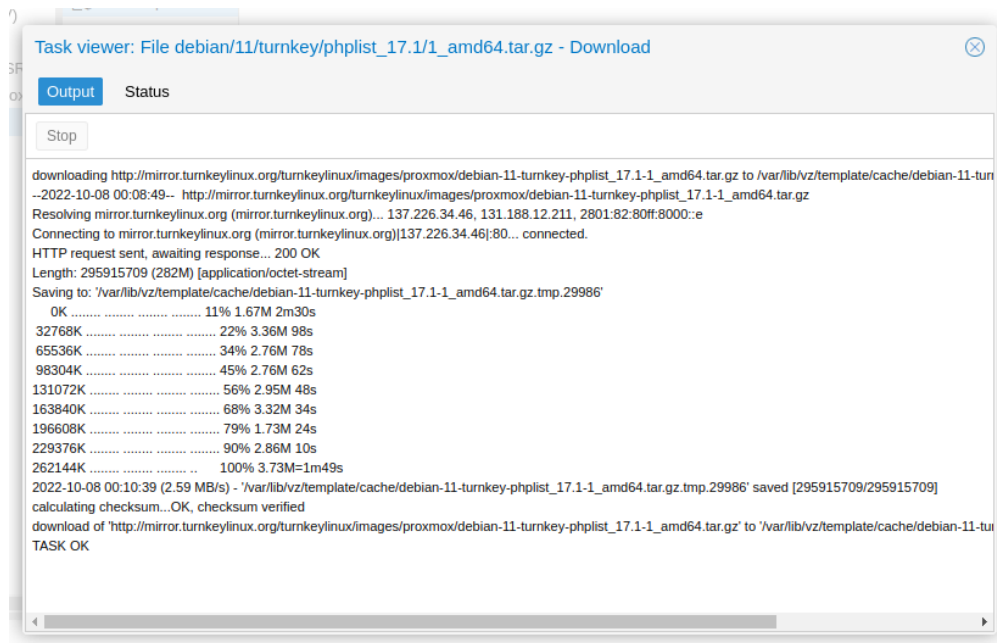


En la ventana emergente, seleccionamos un template y presionamos "Download":

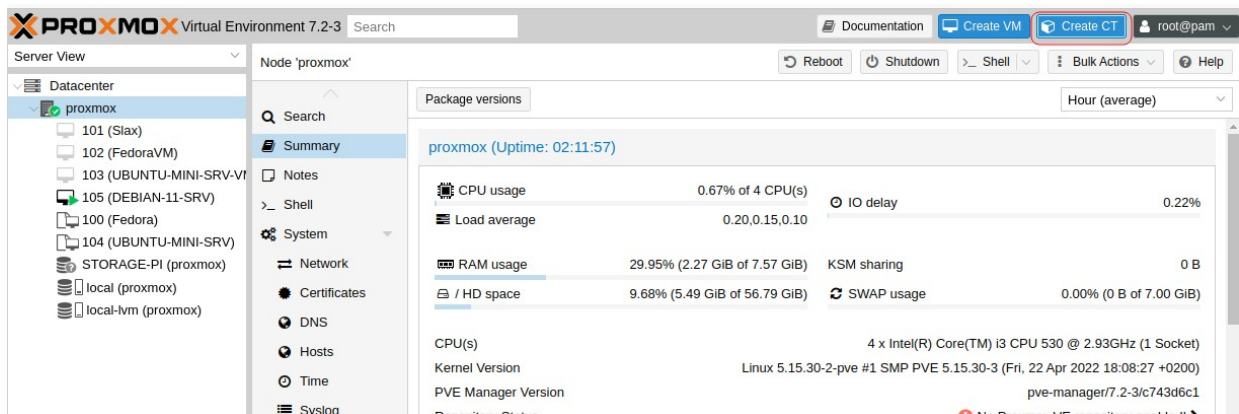


Curso de Virtualización con Proxmox

Al finalizar el proceso correctamente nos indicará con un “Task OK”:



Luego de tener descargado el template en el almacenamiento, comenzaremos a crear el Container, presionamos en “Create CT”:



En el menú desplegable, seleccionamos un nombre para el contenedor, la clave del root y presionamos next:

Create: LXC Container

General Template Disks CPU Memory Network DNS Confirm

Node: proxmox Resource Pool:

CT ID: 106 Password: *****

Hostname: PrimerContenedor Confirm password: *****

Unprivileged container: ☒ SSH public key:

Nesting: ☒ Load SSH Key File

Help Advanced Back Next

En el próximo paso, seleccionamos el storage donde tenemos el template, seleccionamos el template y nos habilita la opción “next”, pulsamos next

Create: LXC Container

General Template Disks CPU Memory Network DNS Confirm

Storage: local

Template:

Name	For...	Size
debian-11-turnkey-phplist_17.1-1_amd64.tar.gz	tgz	295.92 MB

Help Advanced Back Next

Curso de Virtualización con Proxmox

Ahora seleccionamos el storage y asignamos un tamaño de disco:

The screenshot shows the 'Create: LXC Container' wizard in Proxmox VE, specifically the 'Disks' tab. The 'Storage' dropdown menu is set to 'local-lvm' and is highlighted with a red box and the number '1'. The 'Disk size (GiB)' is set to '8' and is also highlighted with a red box and the number '2'. At the bottom right, the 'Next' button is highlighted with a red box and the number '3'. The 'Back' button is also visible. The 'Advanced' checkbox is unchecked. The 'Help' button is on the bottom left.

En la próxima pestaña seleccionamos la asignación de cpu y pulsamos next:

The screenshot shows the 'Create: LXC Container' wizard in Proxmox VE, specifically the 'CPU' tab. The 'Cores' dropdown menu is set to '1' and is highlighted with a red box. At the bottom right, the 'Next' button is highlighted with a red box. The 'Back' button is also visible. The 'Advanced' checkbox is unchecked. The 'Help' button is on the bottom left.

Seleccionamos la cantidad de memoria ram y memoria de intercambio (swap) y pulsamos next:

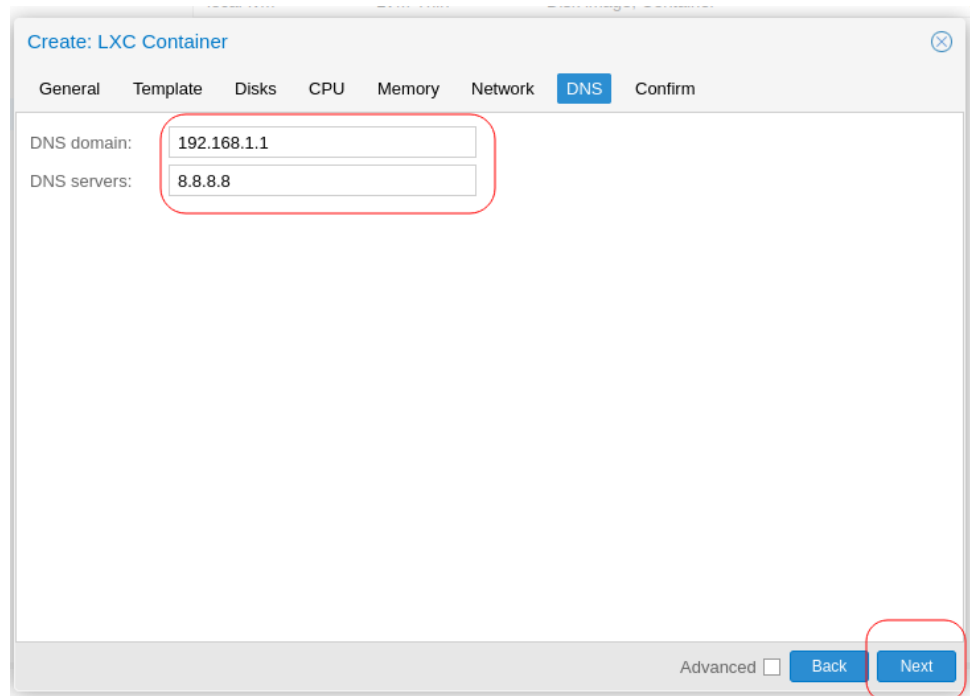
The screenshot shows the 'Create: LXC Container' dialog box with the 'Memory' tab selected. The 'Memory (MiB)' field is set to 512 and the 'Swap (MiB)' field is also set to 512. Both fields are circled in red. At the bottom right, the 'Next' button is also circled in red. The 'Advanced' checkbox is unchecked. The 'Help' button is on the bottom left.

Seleccionamos el nombre del dispositivo de red, el bridge y asignamos una ip (en este caso estática), la máscara y puerta de enlace correspondiente:

The screenshot shows the 'Create: LXC Container' dialog box with the 'Network' tab selected. The 'Name' field is set to 'eth0', the 'Bridge' field is set to 'vbr0', and the 'IPv4' configuration is set to 'Static' with 'IPv4/CIDR' set to '192.168.1.111/24' and 'Gateway (IPv4)' set to '192.168.1.1'. These fields are circled in red. The 'IPv6' configuration is set to 'Static' with 'IPv6/CIDR' set to 'None'. The 'Firewall' checkbox is checked. At the bottom right, the 'Next' button is circled in red. The 'Advanced' checkbox is unchecked. The 'Help' button is on the bottom left.

Curso de Virtualización con Proxmox

Podemos seleccionar un dominio dns y servidores dns, o dejar el campo vacio (toma los valores del host) y pulsamos next:



Create: LXC Container

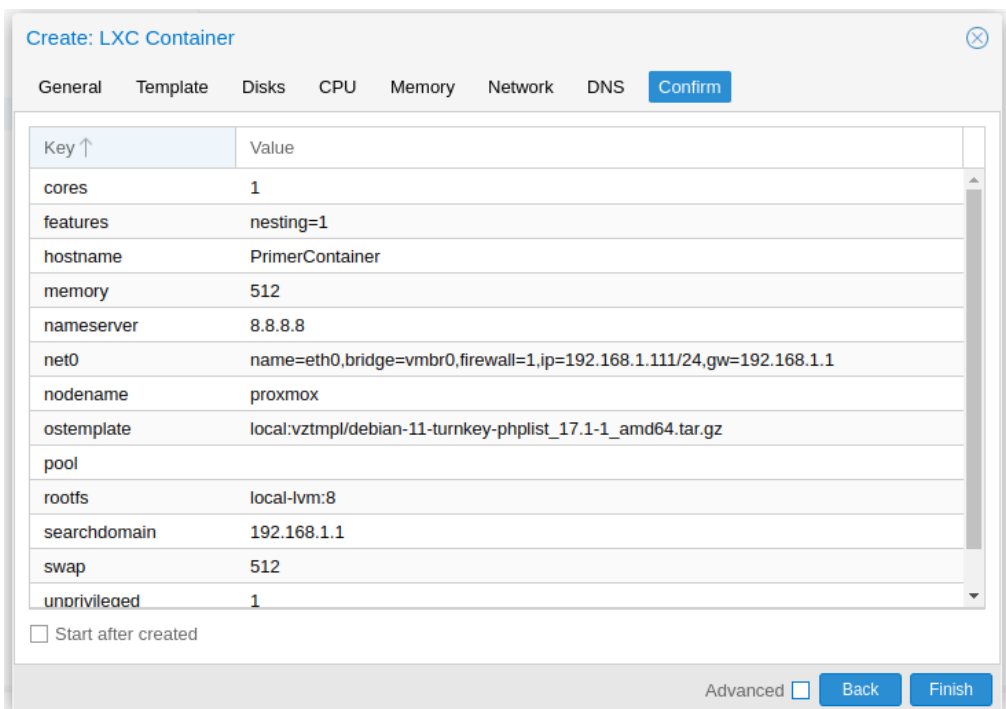
General Template Disks CPU Memory Network **DNS** Confirm

DNS domain: 192.168.1.1

DNS servers: 8.8.8.8

Advanced ☐ Back Next

Para finalizar, se nos muestra un resumen y debemos pulsar en “next”:



Create: LXC Container

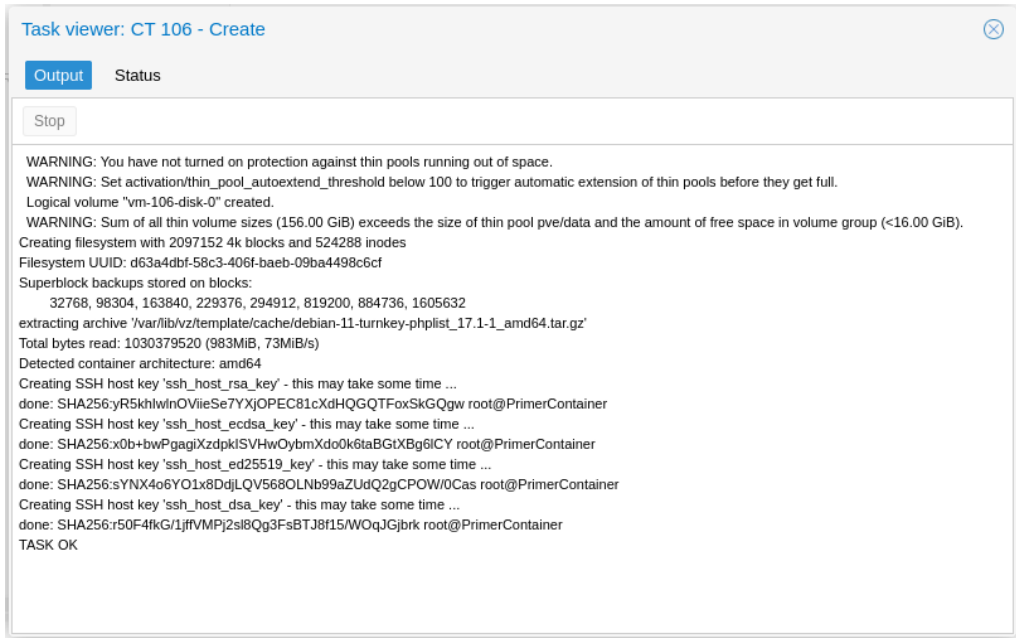
General Template Disks CPU Memory Network DNS **Confirm**

Key ↑	Value
cores	1
features	nesting=1
hostname	PrimerContainer
memory	512
nameserver	8.8.8.8
net0	name=eth0,bridge=vbr0,firewall=1,ip=192.168.1.111/24,gw=192.168.1.1
nodename	proxmox
ostemplate	local:vztmpl/debian-11-turnkey-phplst_17.1-1_amd64.tar.gz
pool	
rootfs	local-lvm:8
searchdomain	192.168.1.1
swap	512
unprivileged	1

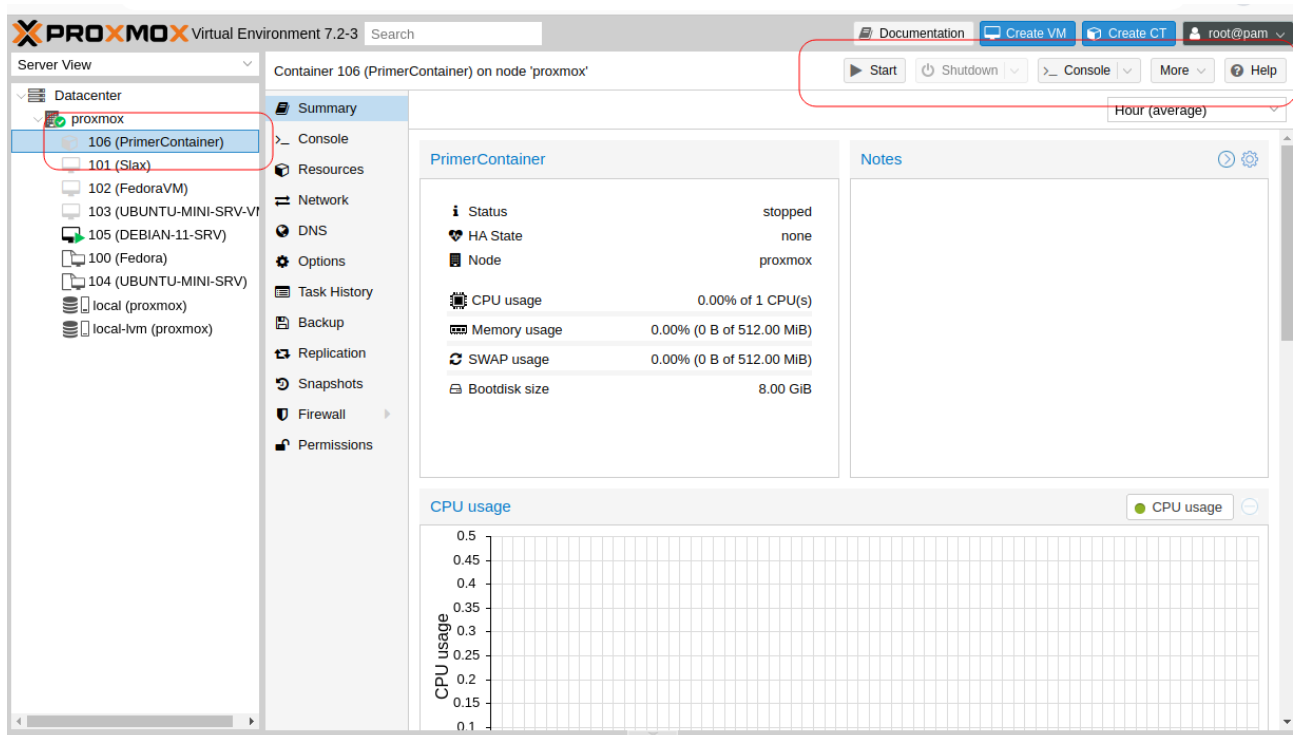
☐ Start after created

Advanced ☐ Back Finish

Una vez finalizado el proceso de creación nos mostrara el mensaje “Task ok” si no se ha producido ninguna excepción:



Y podremos ver el contenedor dentro de nuestro nodo (y así iniciarlo/detenerlo):



Servidor NFS con Debian GNU/Linux

NFS, de Network File System, es un protocolo para acceso remoto a un sistema de archivos mediante una red. Todos los sistemas Unix pueden utilizar este protocolo.

En el kernel de Debian NFS viene incorporado como un módulo, para ejecutar NFS automáticamente al inicio del sistema, debemos instalar el paquete `nfs-kernel-server`, ejecutando el comando:

```
#apt install nfs-kernel-server
```

(requiere permisos root)

Los puertos utilizados serán 111 (para portmapper) y 2049 (NFS)

Adicionalmente, configuraremos la opción “RPCMOUNTDOPTS” en el archivo `/etc/default/nfs-kernel-server` para hacer que el daemon de `rpc.mountd` escuche en el puerto 33333:

```
Comentamos la linea RPCMOUNTDOPTS="--manage-gids -" con un # :  
#RPCMOUNTDOPTS="--manage-gids --port 50003"  
y agregamos:  
RPCMOUNTDOPTS="--port 33333"
```

Sobre `rpc.mountd`: implementa la parte servidor para el montaje `nfs`, para más información sobre `rpc.mountd`:

<https://linux.die.net/man/8/rpc.mountd>

Luego de realizar estos cambios, realizar un reinicio del servicio `nfs`:

```
# service nfs-kernel-server restart
```

Tip: para testear los puertos utilizados por `rpc.mountd`:

```
#rpcinfo -p | grep "tcp.mountd"
```


Permitir NFS a través del firewall

Si disponemos de **UFW** (uncomplicated firewall) debemos abrir los puertos necesarios para permitir nfs.

En caso de no disponer de UFW en Debian, podemos instalarlo con el comando:

```
#apt install ufw
```

Para activar UFW:

```
#ufw enable
```

Ver las reglas de firewall numeradas:

```
#ufw status numbered
```

Reglas necesarias para nfs:

```
#ufw allow in from 192.168.1.10 to any port 111
```

```
#ufw allow in from 192.168.1.10 to any port 2049
```

```
#ufw allow in from 192.168.1.10 to any port 33333
```

En este ejemplo suponemos haber configurado rpc.mountd en el puerto 33333, y el cliente que accederá a NFS se encuentra en 192.168.1.10

Compartir un directorio con NFS

Ahora que disponemos de un servidor NFS, crearemos y compartiremos un directorio.

Creamos el directorio /storage:

```
#mkdir /storage
```

Cambiamos los permisos:

```
#chmod 777 /storage -R
```

Curso de Virtualización con Proxmox

Agregamos el directorio al `/etc/exports`:

```
/storage 192.168.1.10(rw, sync, no_root_squash)
```

Hacemos disponibles los directorios exportados por NFS:

```
#exportfs -ra
```

Si queremos ver los directorios exportados actualmente:

```
#exportfs -s
```

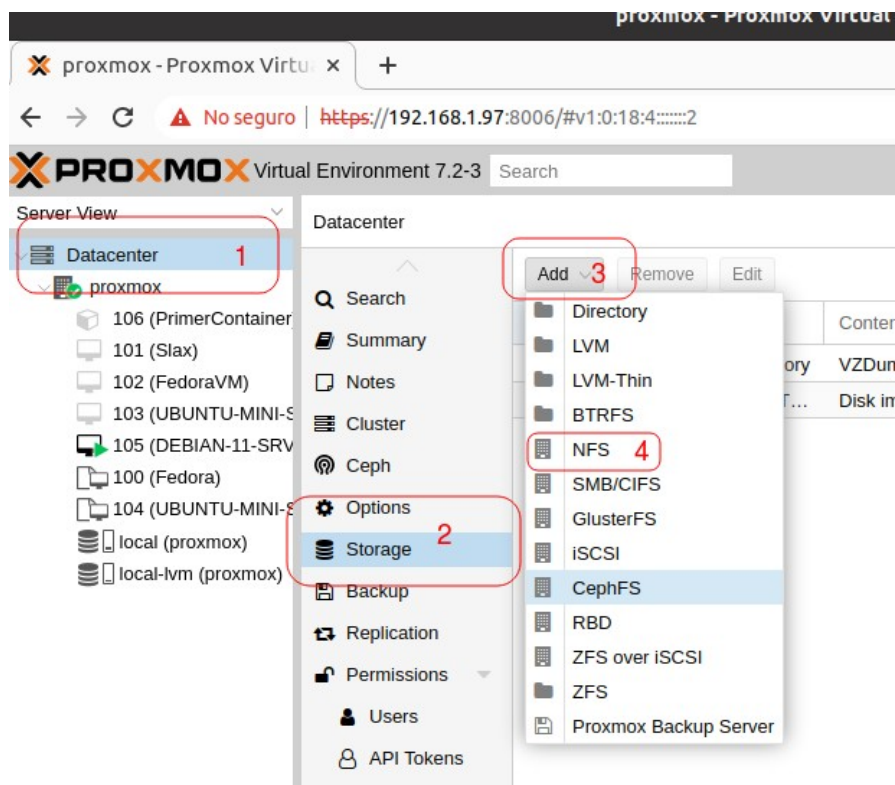
Agregar un Storage NFS a Proxmox

Podemos agregar diferentes tipos de Storage a Proxmox, entre ellos:

- Directory
- LVM
- NFS
- SMB/CIFS
- CephFS
- ZFS
- Proxmox Backup Server

Para agregar un storage NFS:

Seleccionamos el Datacenter, opción Storage, Add y NFS



Luego en la ventana emergente, seleccionamos un nombre para el storage, la dirección ip donde se encuentra, damos clic en el desplegable, seleccionamos el directorio compartido y por último el botón “Add”:

Add: NFS

General Backup Retention

ID: RASP-STORAGE 1

Server: 192.168.1.44 2

Export: /storage-pi 3

Content: /storage-pi 4

Nodes: All (No restrictions)

Enable: ☒

Help Advanced **Add** 5

Tip: por defecto solo se encuentra seleccionada la opción “Disk image”, para permitir otros usos del storage desplegamos las opciones de “Content” y las marcamos antes de pulsar Add:

Add: NFS

General Backup Retention

ID: RASP-STORAGE

Server: 192.168.1.44

Export: /storage-pi

Content: Disk image

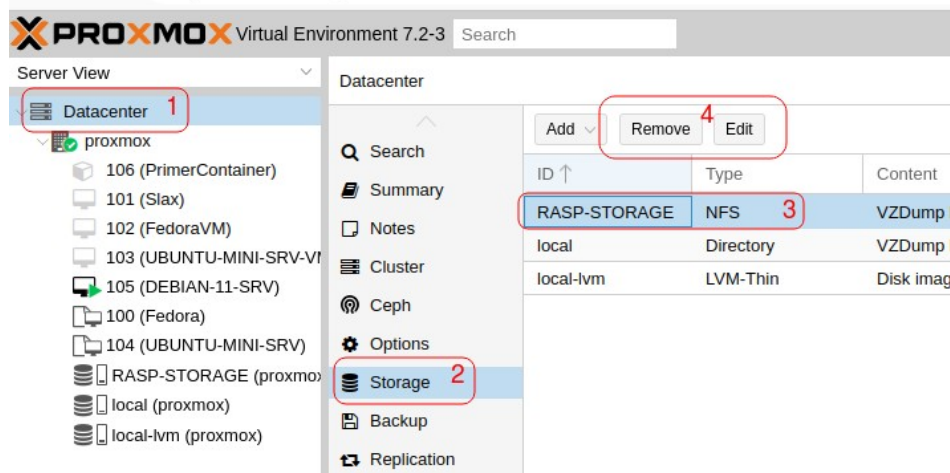
Nodes: All (No restrictions)

Enable: ☒

Help Advanced **Add**

- Disk image
- ISO image
- Container template
- VZDump backup file
- Container
- Snippets

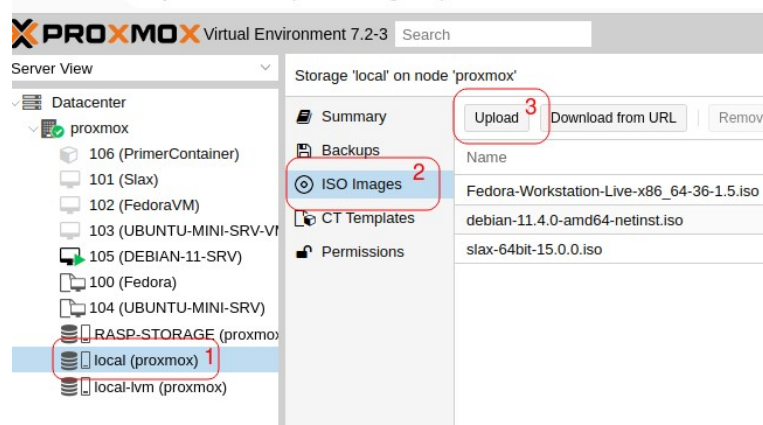
Quitar/Editar un storage: seleccionamos el datacenter, marcamos storage, seleccionamos el storage que queremos alterar y nos habilita los botones “Remove” y “Edit”



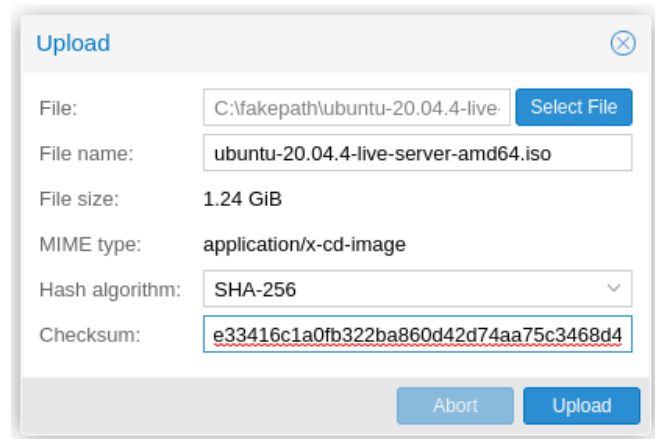
Carga de imágenes ISO hacia Proxmox

Podemos cargar la imagen de instalación de los sistemas operativos que queramos instalar a futuro, desde la interfaz web y con la posibilidad de hacer la comprobación de integridad de dichas imágenes.

Para cargar una imagen iso, seleccionamos un storage que tenga la opción “ISO image”, submenú “ISO image” y presionamos el botón Upload



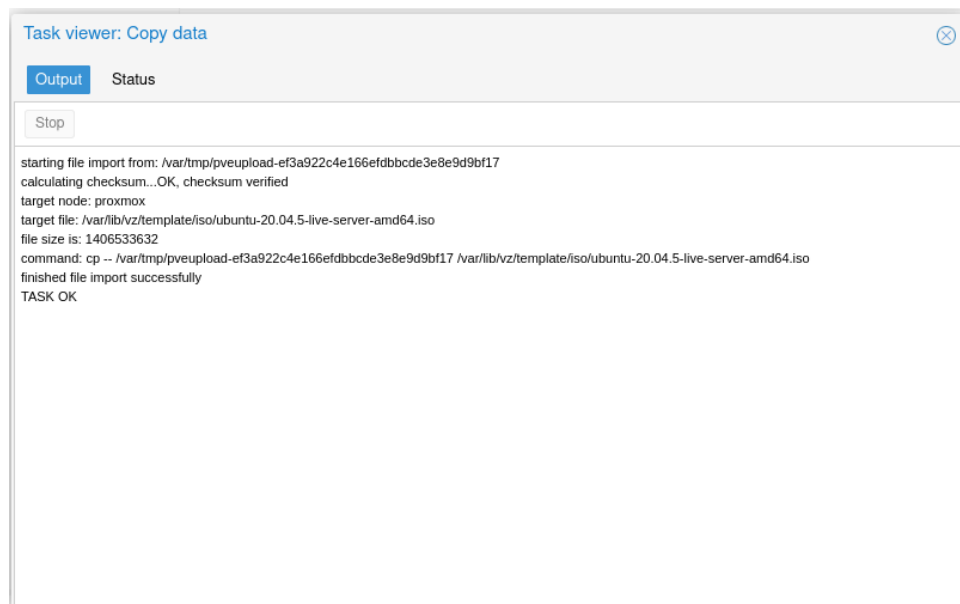
A continuación nos mostrara un menú emergente donde indicamos la podremos buscar la imagen de disco en la ruta local desde la pc con la que accedimos al proxmox, y también podremos pegar la suma de comprobación para chequear la integridad de la imagen:



The screenshot shows the 'Upload' dialog box in Proxmox. It contains the following fields and buttons:

- File:** C:\fakepath\ubuntu-20.04.4-live- (with a 'Select File' button)
- File name:** ubuntu-20.04.4-live-server-amd64.iso
- File size:** 1.24 GiB
- MIME type:** application/x-cd-image
- Hash algorithm:** SHA-256 (dropdown menu)
- Checksum:** e33416c1a0fb322ba860d42d74aa75c3468d4 (highlighted with a red dashed border)
- Buttons:** 'Abort' and 'Upload' at the bottom right.

Luego de pulsar Upload comienza el proceso de subida de la imagen local al servidor, cuando finalice la carga comienza la comprobación de integridad (en caso de ingresar el checksum esperado en el campo checksum), si la comprobación de integridad falla la imagen no se graba en el servidor. En la imagen del ejemplo se muestra la comprobación correcta y resumen de la carga de la imagen:

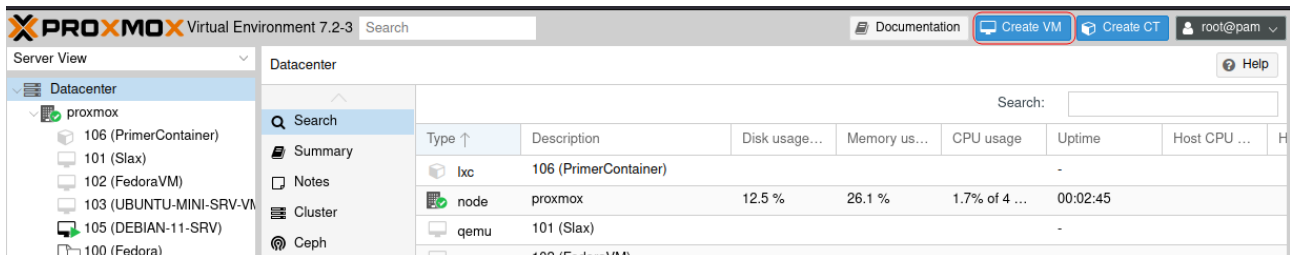


The screenshot shows the 'Task viewer: Copy data' window. It has tabs for 'Output' and 'Status'. The 'Output' tab is active, showing the following log:

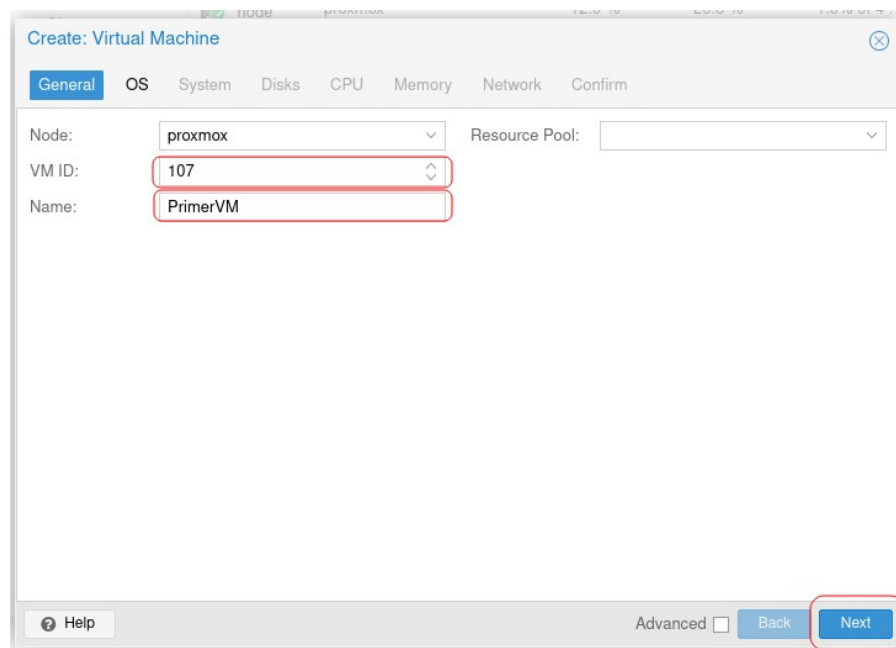
```
starting file import from: /var/tmp/pveupload-ef3a922c4e166efdbbcde3e8e9d9bf17
calculating checksum...OK, checksum verified
target node: proxmox
target file: /var/lib/vz/template/iso/ubuntu-20.04.5-live-server-amd64.iso
file size is: 1406533632
command: cp -- /var/tmp/pveupload-ef3a922c4e166efdbbcde3e8e9d9bf17 /var/lib/vz/template/iso/ubuntu-20.04.5-live-server-amd64.iso
finished file import successfully
TASK OK
```

Creación de una Maquina Virtual

Para crear una maquina virtual presionamos el botón “Create VM” en la parte superior sobre el lateral derecho:

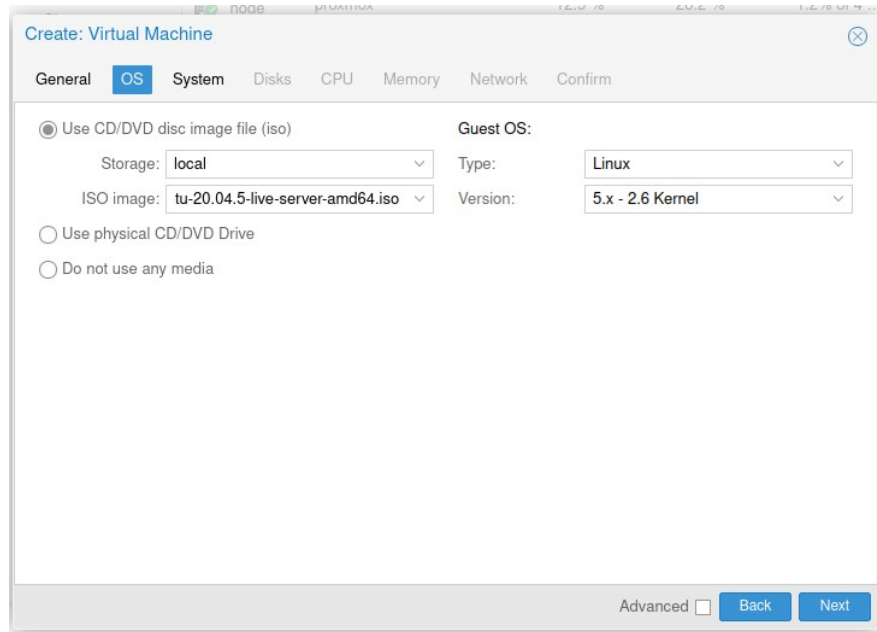


Luego se nos preguntará el ID y Nombre de la VM, luego de completar esos datos pulsamos next:



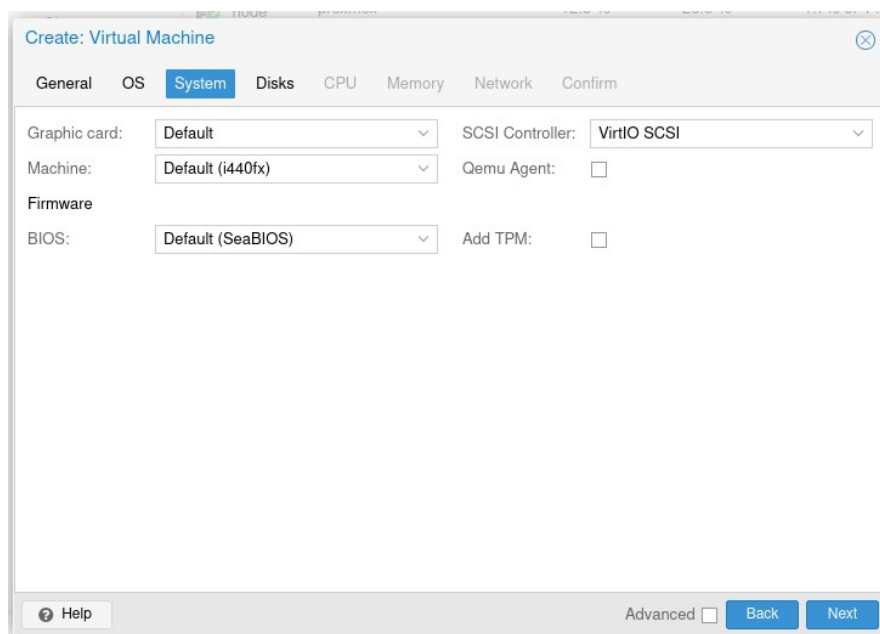
Curso de Virtualización con Proxmox

Ahora debemos seleccionar un storage que contenga la imagen iso de instalación, el tipo de sistema (Linux, Windows, Solares, otro) seleccionamos la imagen de instalación y pulsamos next:



The screenshot shows the 'Create: Virtual Machine' window with the 'OS' tab selected. The 'Use CD/DVD disc image file (iso)' option is chosen. The 'Storage' is set to 'local' and the 'ISO image' is 'tu-20.04.5-live-server-amd64.iso'. Under 'Guest OS', the 'Type' is 'Linux' and the 'Version' is '5.x - 2.6 Kernel'. There are also options for 'Use physical CD/DVD Drive' and 'Do not use any media'. At the bottom, there is an 'Advanced' checkbox and 'Back' and 'Next' buttons.

En este paso podemos seleccionar la tarjeta grafica, chipset (i440fx/q35), tipo de BIOS y controladora de disco, luego pulsamos next:



The screenshot shows the 'Create: Virtual Machine' window with the 'System' tab selected. The 'Graphic card' is set to 'Default', the 'Machine' is 'Default (i440fx)', and the 'BIOS' is 'Default (SeaBIOS)'. The 'SCSI Controller' is set to 'VirtIO SCSI'. There are checkboxes for 'Qemu Agent' and 'Add TPM'. At the bottom, there is a 'Help' button, an 'Advanced' checkbox, and 'Back' and 'Next' buttons.

Curso de Virtualización con Proxmox

En este paso (Disks) se nos solicita el tipo de bus (IDE, SATA, VirtIO Block o SCSI), el storage donde se almacenara la unidad de disco y el tamaño en Giga Bytes de la unidad de disco para la vm. Podemos agregar más de una unidad, luego pulsamos next:

The screenshot shows the 'Create: Virtual Machine' window with the 'Disks' tab selected. The configuration for the 'virtio0' disk is as follows:

Field	Value
Bus/Device:	VirtIO Block
Storage:	local-lvm
Disk size (GiB):	32
Format:	Raw disk image (raw)
Cache:	Default (No cache)
Discard:	<input type="checkbox"/>

At the bottom left, there is an 'Add' button. At the bottom right, there are 'Back' and 'Next' buttons, and an 'Advanced' checkbox.

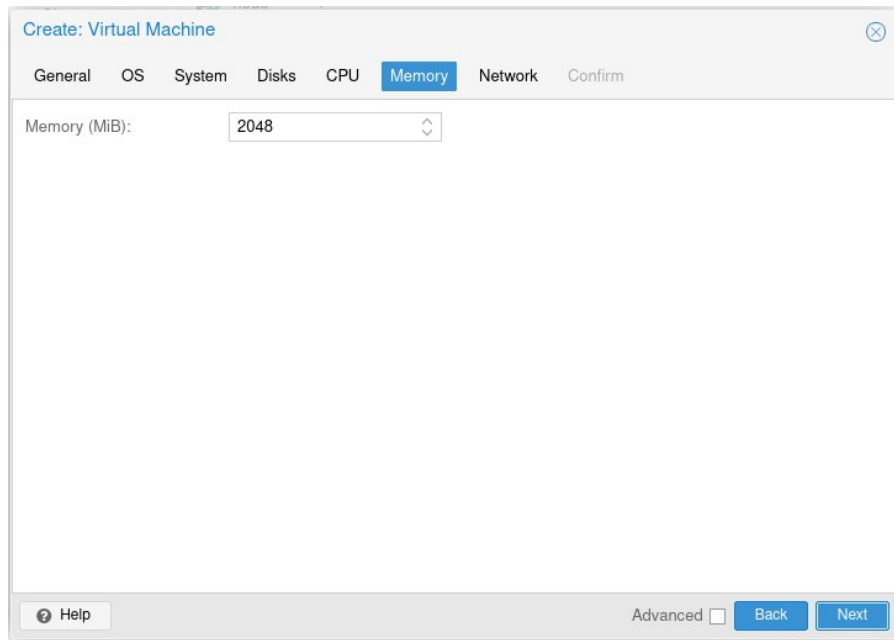
Luego se nos preguntara por la cantidad de sockets y cpu asignadas a esta VM, y pulsamos next:

The screenshot shows the 'Create: Virtual Machine' window with the 'CPU' tab selected. The configuration for the CPU is as follows:

Field	Value
Sockets:	1
Cores:	2
Total cores:	2
Type:	Default (kvm64)

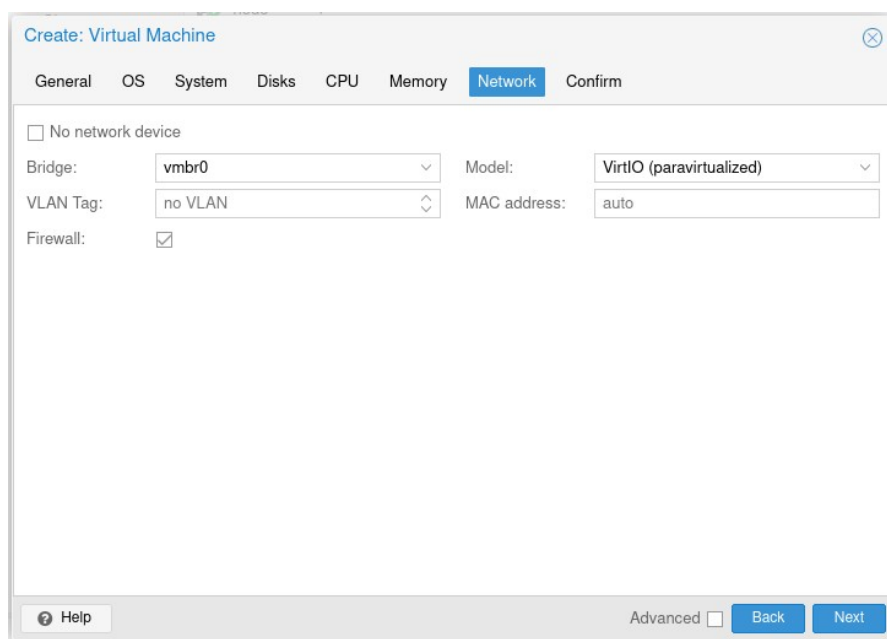
At the bottom right, there are 'Back' and 'Next' buttons, and an 'Advanced' checkbox.

Ahora se nos pregunta la cantidad de memoria RAM asignada, y pulsamos next:



The screenshot shows the 'Create: Virtual Machine' dialog box with the 'Memory' tab selected. The 'Memory (MiB)' field is set to 2048. The tabs at the top are General, OS, System, Disks, CPU, Memory, Network, and Confirm. The bottom of the dialog has a Help button, an Advanced checkbox, and Back and Next buttons.

En este paso se nos solicita lo relativo a la red; podemos seleccionar un bridge diferente en caso de estar configurado previamente, el modelo de tarjeta de red para la VM, si se encontrará taggeada en una VLAN determinada, o si la VM no debe tener interface de red, luego pulsamos next:



The screenshot shows the 'Create: Virtual Machine' dialog box with the 'Network' tab selected. The 'No network device' checkbox is unchecked. The 'Bridge' dropdown is set to 'vmbr0', the 'Model' dropdown is set to 'VirtIO (paravirtualized)', the 'VLAN Tag' dropdown is set to 'no VLAN', and the 'MAC address' field is set to 'auto'. The 'Firewall' checkbox is checked. The tabs at the top are General, OS, System, Disks, CPU, Memory, Network, and Confirm. The bottom of the dialog has a Help button, an Advanced checkbox, and Back and Next buttons.

Curso de Virtualización con Proxmox

Llegados a este punto nos mostrará un resumen de las opciones seleccionadas, pulsamos Finish para iniciar la creación de la VM:

Key ↑	Value
cores	2
ide2	local:iso/ubuntu-20.04.5-live-server-amd64.iso,media=cdrom
machine	q35
memory	2048
name	PrimerVM
net0	virtio,bridge=vbr0,firewall=1
nodename	proxmox
numa	0
ostype	l26
scsihw	virtio-scsi-pci
sockets	1
virtio0	local-lvm:32
vmid	107

☐ Start after created

Advanced ☐ Back Finish

Al finalizar, se muestra un status ok en el listado de tareas, la VM aparecerá dentro del nodo y podremos iniciarla con el botón "start":

Proxmox Virtual Environment 7.2-3

Server View: proxmox

- 106 (PrimerContainer)
- 101 (Slax)
- 102 (FedoraVM)
- 103 (UBUNTU-MINI-SRV-VM)
- 105 (DEBIAN-11-SRV)
- 107 (PrimerVM)**
- 100 (Fedora)
- 104 (UBUNTU-MINI-SRV)
- RASP-STORAGE (proxmox)
- local (proxmox)
- local-lvm (proxmox)

VirtualBox

Virtual Machine 107 (PrimerVM) on node 'proxmox'

Start Shutdown Console More Help

Hour (average)

PrimerVM

Status: stopped

HA State: none

Node: proxmox

CPU usage: 0.00% of 2 CPU(s)

Memory usage: 0.00% (0 B of 2.00 GiB)

Bootdisk size: 32.00 GiB

IPs: No Guest Agent configured

CPU usage graph

Tasks

Start Time	End Time	Node	User name	Description	Status
Oct 10 21:54:06	Oct 10 21:54:07	proxmox	root@pam	VM 107 - Create	OK
Oct 10 21:27:52	Oct 10 21:27:59	proxmox	root@pam	Start all VMs and Containers	OK
Oct 10 21:27:52	Oct 10 21:27:59	proxmox	root@pam	VM 105 - Start	OK

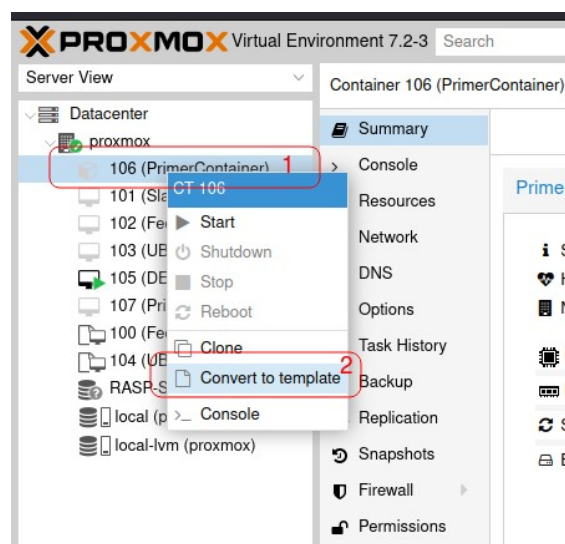
Creación de Templates en Proxmox

Podemos crear plantillas (Templates) que podríamos llamar “moldes” con los cuales crear replicas de las maquinas en un estado deseado, con ello podríamos instalar un sistema, configurar lo para nuestras necesidades y luego crear copias con los recursos ya disponibles (por ejemplo, servidores LAMP con determinada configuración, y cada vez que necesitamos un nuevo servidor clonamos ese template y obtenemos un nuevo servidor con todas las características del template).

Esto nos permite ahorrar tiempo de instalación y configuración.

Podemos crear templates tanto de las VM como de los containers, siendo el mismo proceso pero aplicado a uno u otro tipo de virtualización.

Pasos para crear un template (ejemplo con un container):
seleccionamos el container con botón derecho y “Convert to template”:



Importante: Los templates no se pueden iniciar como una VM/Container, y tener en cuenta que al convertirlo a template no lo tendremos disponible hasta que hagamos un “clon” de dicho template.

Curso de Virtualización con Proxmox

Luego se nos mostrará un menú emergente donde debemos confirmar pulsando en el botón “Yes”:



Luego de terminada la creación del template a partir del container, cambia el icono del container a icono de template y nos muestra que el estado de la tarea de creación de template termino con status ok.

Además desaparecen los iconos de Start, Shutdown y Console (ya que los templates no se pueden iniciar, solo clonar):

The screenshot shows the Proxmox Virtual Environment 7.2-3 web interface. On the left, the "Server View" sidebar shows a tree of resources under "Datacenter" > "proxmox". "106 (PrimerContainer)" is highlighted with a red box. The main panel shows the details for "Container 106 (PrimerContainer) on node 'proxmox'". The "Summary" tab is active, displaying resource information: HA State (none), Node (proxmox), Processors (1), Memory (512.00 MiB), Swap (512.00 MiB), and Bootdisk size (8.00 GiB). Below this is a "Notes" section. At the bottom, the "Tasks" tab is selected, showing a table of recent tasks. The first task is highlighted with a red box.

Start Time ↓	End Time	Node	User name	Description	Status
Oct 10 22:26:59	Oct 10 22:27:00	proxmox	root@pam	CT 106 - Convert to template	OK
Oct 10 21:54:06	Oct 10 21:54:07	proxmox	root@pam	VM 103 - Create	OK

Creación de cuentas de Usuario

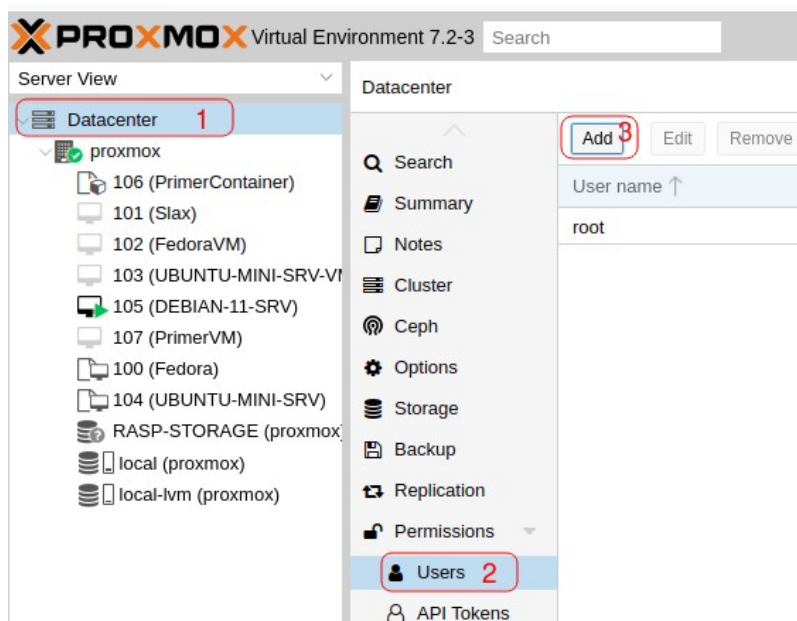
Proxmox provee varios métodos de autenticación, entre ellos:

- Linux PAM
- Proxmox VE Authentication Server
- LDAP
- Microsoft Active Directory
- OpenID Connect

La autenticación Linux PAM corresponde a usuarios del sistema host, debe existir en primera instancia un usuario de sistema en cada nodo para que se pueda utilizar este tipo de autenticación (se debe agregar el usuario con adduser desde la terminal del sistema host)

Creación de un usuario PAM:

Seleccionamos el Datacenter, menu “Users” y clic en el botón Add:

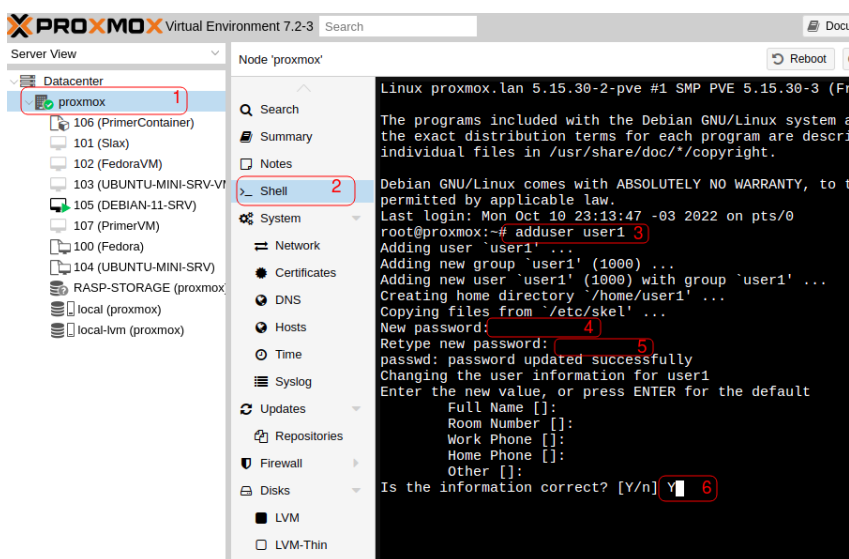


Se nos presentará el siguiente menú:

Escribimos el nombre de usuario (en el ejemplo “user1”), seleccionamos “Linux PAM standard authentication” y pulsamos el botón “Add”.

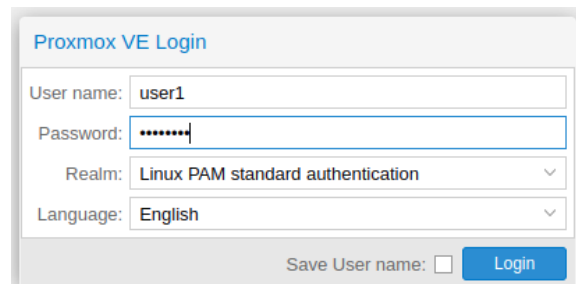
Ahora debemos crear el usuario como usuario del sistema host, para lo cual seleccionamos el nodo, Shell y ejecutamos el comando adduser (adduser user1 en el ejemplo; es importante que el nombre de usuario que ingresamos por consola coincida con el usuario que fue creado desde la interfaz web).

Luego se nos pedirá un password, repetir el password, la información personal que se nos solicita podemos obviarla pulsando enter, luego nos solicita confirmar con la tecla Y (Yes) y enter:



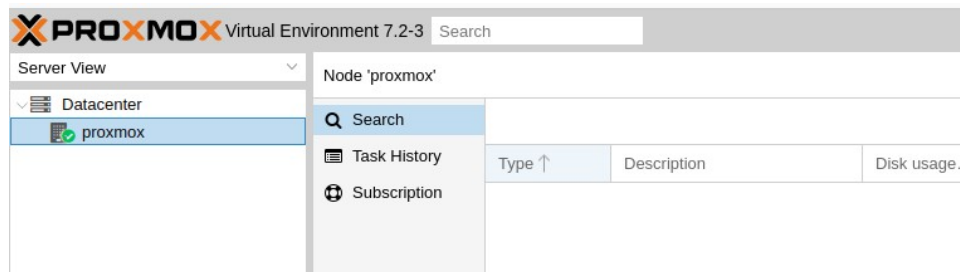
Curso de Virtualización con Proxmox

Hemos creado el usuario, para corroborar su funcionamiento hacemos login con estas nuevas credenciales:



The image shows the 'Proxmox VE Login' dialog box. It contains the following fields and options:

- User name:
- Password:
- Realm:
- Language:
- Save User name: ☐
- Login button



Como se aprecia, si bien hemos creado el usuario correctamente (usuario del sistema host) no tenemos ningún permiso para hacer uso de Proxmox!

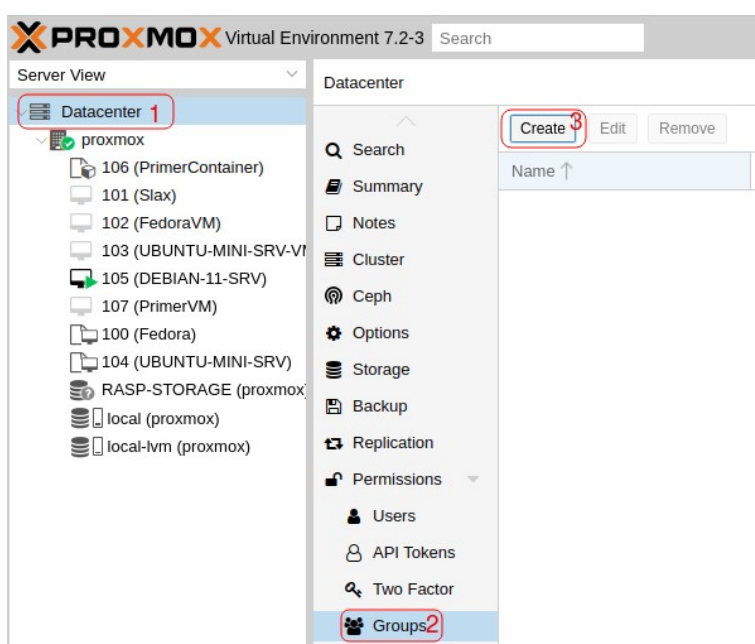
Tanto si creamos un usuario con autenticación de tipo PAM o PVE, debemos asignar permisos para que dicho usuario pueda realizar tareas.

La diferencia mas notoria es que con un usuario de tipo PAM podemos hacer login por medio de SSH.

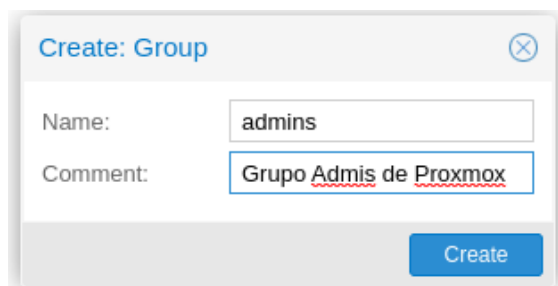
Grupos y asignación de Permisos a usuarios

Crearemos un grupo de usuarios, al cual podremos asignarle permisos. Luego haremos que un usuario sea parte de ese grupo, por tanto “heredará” los permisos del grupo.

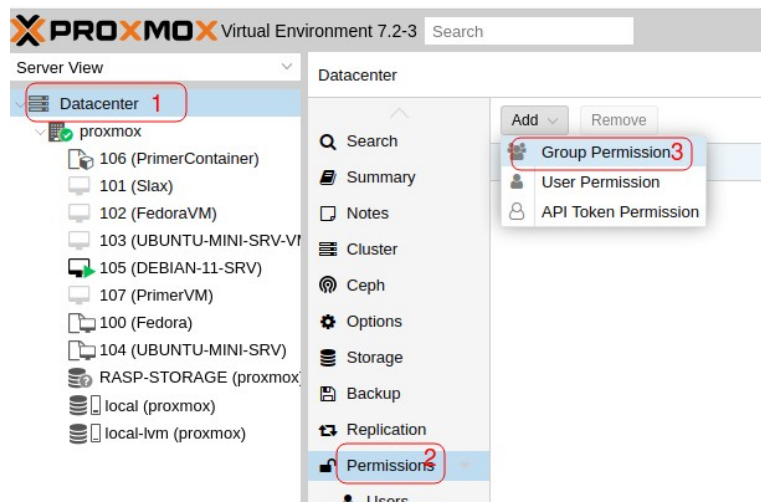
Damos clic en el Datacenter, “Groups” y “Create”



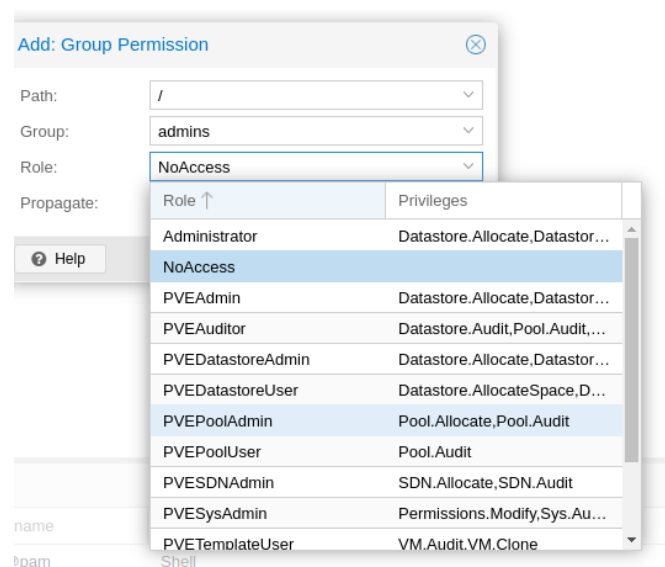
Creamos el grupo “admins” y le damos un comentario:



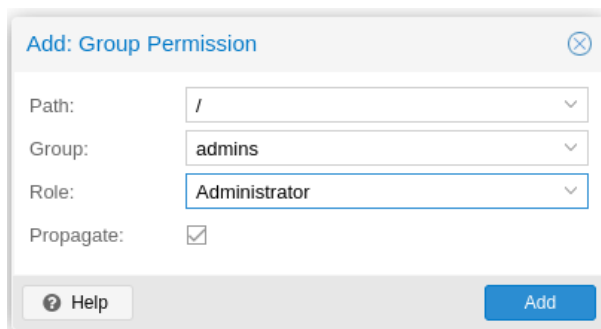
Luego de creado el grupo, asignaremos permisos a dicho grupo, damos clic en el Datacenter, “Permissions” y en el botón Add seleccionamos “Group Permission”:



Se nos abrirá un menú desplegable, en el cual podemos seleccionar diferentes niveles de permisos, tener en cuenta que el primer nivel (símbolo slash, /) indica “todos los permisos”; podríamos seleccionar permisos solo a nivel de un storage, a nivel de un nodo o a nivel de una VM, entre otros (seleccionaremos / para el ejemplo del grupo admins). Además debemos seleccionar un Rol (en el ejemplo rol Administrator):



Una vez seleccionado nivel, grupo y rol pulsamos “Add”

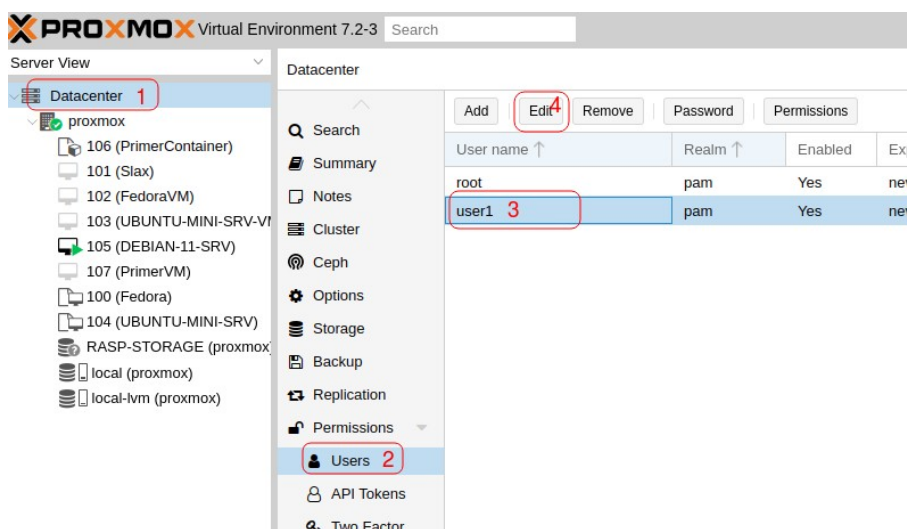


The dialog box titled "Add: Group Permission" contains the following fields:

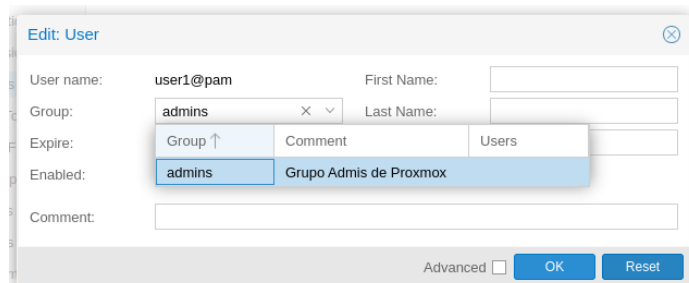
- Path: /
- Group: admins
- Role: Administrator
- Propagate: ☒

Buttons: Help, Add

Ahora nos encontramos en condiciones de asignar un grupo con ciertos permisos, a uno o varios usuarios. Seleccionamos el datacenter, Users, seleccionamos el usuario que queremos modificar y “Edit”:



Ahora en “Group” desplegamos y seleccionamos el grupo “admins” creado previamente, y pulsamos “ok”



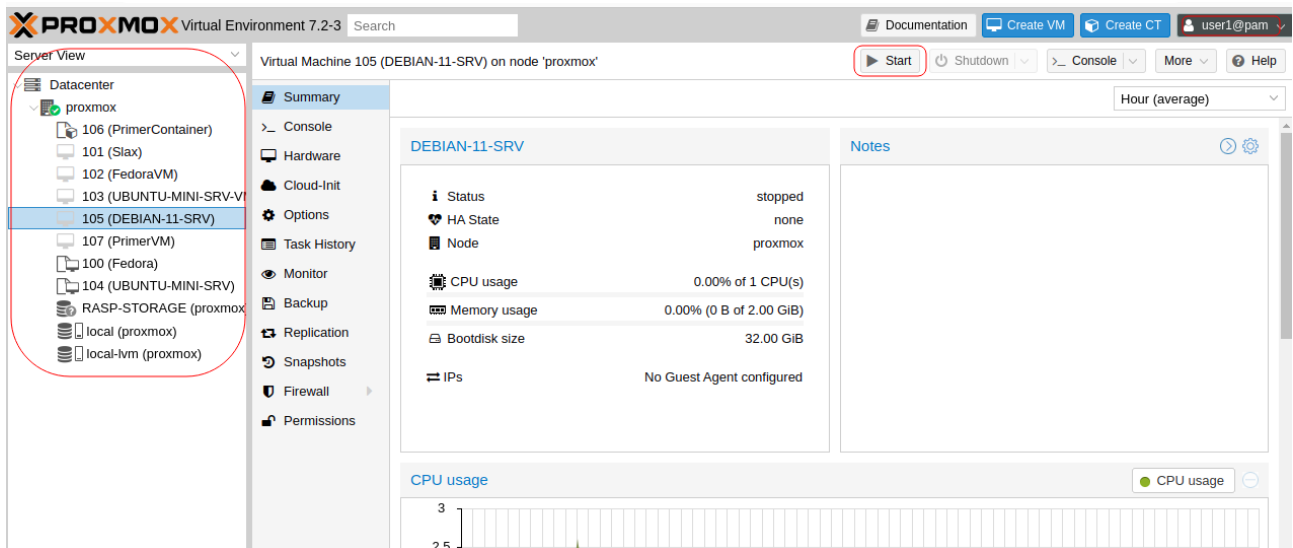
The dialog box titled "Edit: User" contains the following fields:

- User name: user1@pam
- First Name:
- Group: admins (selected from a dropdown menu)
- Last Name:
- Expire:
- Comment:
- Enabled: ☒

Buttons: OK, Reset

Curso de Virtualización con Proxmox

Una vez asignado el grupo al usuario (y con ello los permisos asociados a ese grupo), hacemos login en proxmox, observemos que ahora el usuario “user1” puede controlar las VM y otras opciones del nodo:

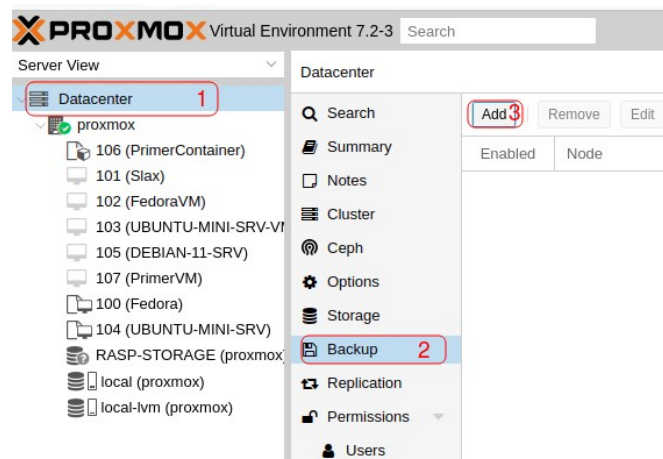


Si bien es posible crear usuarios de esta manera (autenticación PAM) es aconsejable utilizar para la interfaz web, usuarios de tipo Proxmox VE Authentication Server (pve), y crear grupos y asignar roles en forma más granular (por ejemplo, podemos tener grupos de usuarios para algunas vm, o algunos storage)

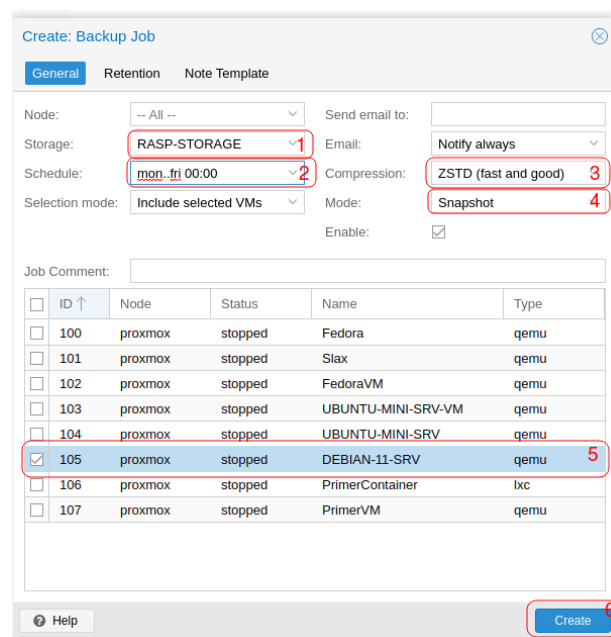
Automatización de Backups

Es importante contar con respaldos de nuestras VM o Containers. Proxmox nos permite programar BACKUP's automatizados.

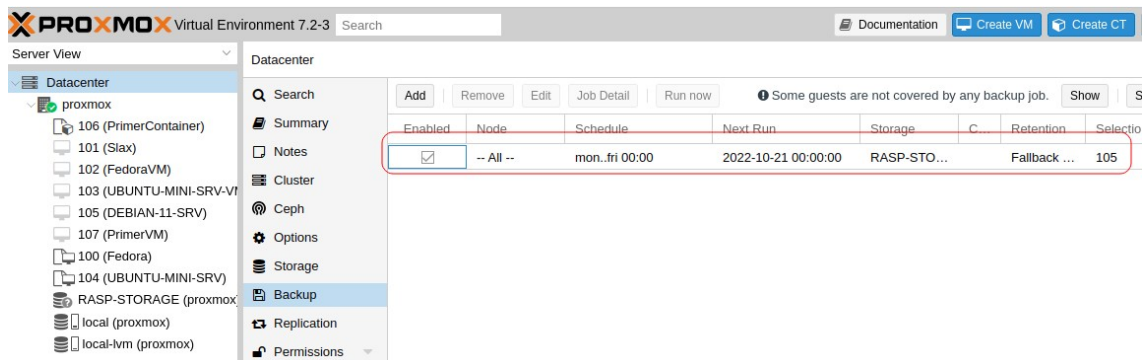
Para crear un backup automatizado, seleccionamos el Datacenter, "Backup" y presionamos "Add":



En el menú contextual seleccionamos el storage donde se guardarán los backups, la agenda (mensual, semanal, etc) el tipo de compresión, la o las VM y pulsamos "Create"



Una vez agendado el backup, podremos verlo en el listado:



Copia de una VM desde una unidad usb

Podemos copiar una VM contenida en una unidad USB a un nodo de proxmox, los pasos son los siguientes:

1: Buscar la unidad con fdisk:

```
#fdisk -l
```

2: Crear una carpeta en /media/ (por ejemplo: /media/usb)

```
#mkdir /media/usb
```

3: montar la unidad en el directorio creado:

```
#mount /dev/sdb /media/usb
```

4: Copiar desde la unidad usb a la carpeta local de los backups:

```
// Copia de la imagen:
```

```
#cp /media/usb/mi_bkp_de_vm.vma.zst /var/lib/vz/dump/
```

```
// Copia de las notas de la imagen:
```

```
#cp /media/usb/mi_bkp_de_vm.vma.zst.notes /var/lib/vz/dump/
```

```
// Copia de log de creacion de la imagen, opcional:
```

```
#cp /media/usb/mi_bkp_de_vm.log /var/lib/vz/dump/
```

Ahora podemos levantar la VM a partir del backup, que se encontrará copiado en el almacenamiento local del nodo de proxmox. En forma análoga, también podemos copiar una imagen de backup local a un almacenamiento usb para trasladar a otro equipo.

Protección de Proxmox mediante fail2ban

Fail2ban es una aplicación creada para prevenir la intrusión de sistemas, mediante el análisis de archivos de log, procesando los mismos mediante el uso de filtros, y enviando los equipos que cumplan con dichos filtros a “jaulas” (jail’s).

Cada ip que ha sido enviada a una jaula, es bloqueada y no se le permite el acceso al servidor.

Fail2ban tiene un componente cliente, el cual permite ver el estado de las jaulas, y “des-banear” (desbloquear) una ip, volviendo a permitir así el acceso.

Podemos crear jaulas para diferentes servicios: ssh, apache, nginx, mysql, proxmox, entre otros.

Podemos ver una lista de filtros listando los archivos en /etc/fail2ban/filter.d/

Pasos para la instalación y configuración de fail2ban en Proxmox:

1. Instalar fail2ban:

```
#apt install fail2ban
```

2. Modificamos el archivo etc/fail2ban/jail.d/defaults-debian.conf

```
[proxmox]
enabled = true
port = https,http,8006
filter = proxmox
logpath = /var/log/daemon.log
maxretry = 3
bantime = 120
```

3. Crear un archivo de filtro: etc/fail2ban/filter.d/proxmox.conf

```
[Definition]
failregex = pvedaemon\[.*authentication failure; rhost=<HOST>
user=.* msg=.*
ignoreregex =
```

4. Para testear el filtro, realizar un intento de login desde la interfaz web a proxmox con un usuario/clave no valido, y ejecutar:

```
#fail2ban-regex /var/log/daemon.log
/etc/fail2ban/filter.d/proxmox.conf
```

5. Reiniciar fail2ban:

```
#systemctl restart fail2ban
```

Con los pasos anteriores, protegeremos Proxmox de accesos no validos a la interface web (https, http, 8006), además es conveniente proteger también los accesos por ssh (puerto 22)

Para habilitar la protección de accesos por ssh, los pasos son los siguientes:

1. Modificamos nuevamente el archivo

/etc/fail2ban/jail.d/defaults-debian.conf

```
[sshd]
enabled = true
maxretry = 3
bantime = 120m
```

2. Reiniciar fail2ban:

```
#systemctl restart fail2ban
```


Para verificar el estado de las jaulas:

Ejemplo jaula sshd:

```
#fail2ban-client status sshd
```

Ejemplo jaula proxmox:

```
#fail2ban-client status proxmox
```

Resultado (ejemplo jaula proxmox):

```
Status for the jail: proxmox
|- Filter
|  |- Currently failed: 0
|  |- Total failed: 0
|  `-- File list:      /var/log/daemon.log
`-- Actions
    |- Currently banned: 0
    |- Total banned: 0
    `-- Banned IP list:
```

Para des-banear un cliente(quitar una ip de la lista de bloqueos para una jaula, en el ejemplo ssh y cliente en ip 192.168.1.105):

```
#fail2ban-client set ssh unbanip 192.168.1.105
```