

# **REDES**



# **PROXMOX**

IVANA SÁNCHEZ PÉREZ

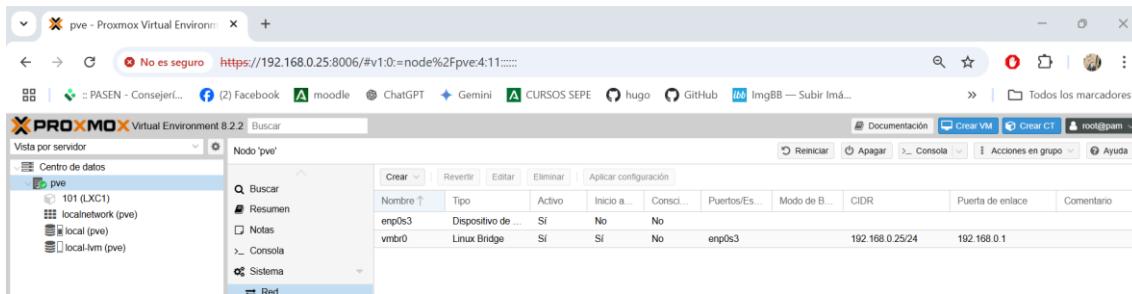
2º ASIR

## Contenido

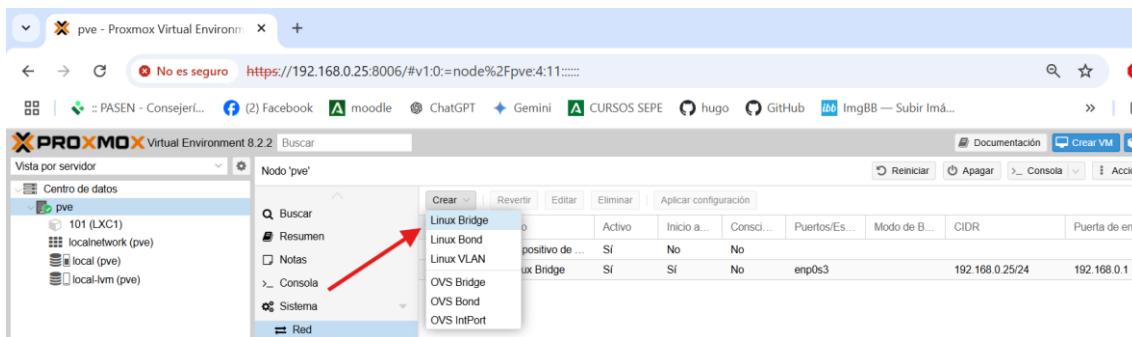
<b>1.</b>	<b>Creación de un bridge .....</b>	<b>3</b>
<b>2.</b>	<b>Creación nueva máquina virtual .....</b>	<b>4</b>
<b>3.</b>	<b>Configuración de la interfaz .....</b>	<b>6</b>
<b>4.</b>	<b>Creación de un contenedor LXC.....</b>	<b>7</b>
<b>5.</b>	<b>Comprobación conectividad.....</b>	<b>8</b>
<b>6.</b>	<b>Instala en la máquina router un servidor web.....</b>	<b>9</b>
<b>7.</b>	<b>Activa el cortafuegos para la interfaz que está conectada a vmbr0. .....</b>	<b>10</b>
<b>8.</b>	<b>Comprueba que no tenemos acceso al servidor web desde el exterior... </b>	<b>12</b>
<b>9.</b>	<b>Crea una regla de cortafuego de entrada para que sea accesible el puerto 80 del protocolo TCP. .....</b>	<b>12</b>
<b>10.</b>	<b>Comprueba que ya tenemos acceso al servidor web desde el exterior</b>	<b>13</b>

# 1. Creación de un bridge

Partimos de mi configuración inicial que por defecto utiliza el router físico, el puente vmbr0, para que se conecten tanto las máquinas virtuales como los contenedores.



Para crear el nuevo bridge, elegimos la opción **System → Network → Create → Linux Bridge**.



Se abrirá una ventana en la que configuraremos los parámetros que se proporcionan de la nueva conexión. Luego aplicamos los cambios con **Apply Configuration** para que se hagan efectivos.

**Crear: Linux Bridge**

Nombre:	vmbr100	Inicio automático:	<input checked="" type="checkbox"/>
IPv4/CIDR:	10.0.0.100/24	Consciente de VLAN:	<input type="checkbox"/>
Puerta de enlace (IPv4):		Puertos de puente:	
IPv6/CIDR:		Comentario:	
Puerta de enlace (IPv6):			
MTU:	1500		

**Ayuda** Avanzado  **Crear**

Nombre	Tipo	Activo	Inicio a...	Consci.	Puertos/Es...	Modo de B...	CIDR	Puerta de enlace	Comentario
emp0s3	Dispositivo de red	Sí	No	No					
vmbr0	Linux Bridge	Sí	Sí	No	emp0s3		192.168.0.25/24	192.168.0.1	
<b>vmbr100</b>	<b>Linux Bridge</b>	<b>Sí</b>	<b>Sí</b>	<b>No</b>			<b>10.0.0.100/24</b>		

## 2. Creación nueva máquina virtual

Ahora procedemos a crear una nueva VM a la que llamaremos router, para ello utilizaré una plantilla de Debian 12 que tengo creada de un ejercicio anterior.

Nombre	Fecha	Formato	Tamaño
debian-12-standard_12.7-1_amd64.tar.zst	2024-12-07 12:15:51	tzst	126.52 MB

Y creamos nuestro contenedor pinchando en **añadir CT** y configurándolo con el nuevo puente creado.

Crear: Contenedor LXC

General Plantilla Discos CPU Memoria Red DNS **Confirmar**

Key ↑	Value
cores	2
features	nesting=1
hostname	router
memory	2048
net0	name=eth0,bridge=vmbr100,firewall=1
nodename	pve
ostemplate	local:vztmp/debian-12-standard_12.7-1_amd64.tar.zst
pool	
rootfs	local-lvm:25
ssh-public-keys	
swap	2048
unprivileged	1
vmid	100

Iniciar después de la creación

**Avanzado**  **Atrás** **Finalizar**

X PROXMOX Virtual Environment 8.2.2 | Buscar

Vista por servidor | Almacenamiento 'local' en el nodo 'pve'

- Centro de datos
  - pve
    - 100 (router) **✓**
    - 101 (LXC1)
    - localnetwork (pve)
    - local (pve)
    - local-lvm (pve)

Resumen | Cargar | Descargar desde URL | Plantillas | Eliminar

**Task viewer: CT 100 - Crear**

**Salida** Estado

Parar | Descargar

```

WARNING: You have not turned on protection against thin pools running out of space.
WARNING: Set activation/thin_pool_autoextend_threshold below 100 to trigger automatic extension of thin pools before they get full.
Logical volume "vm-100-disk-0" created.
WARNING: Sum of all thin volume sizes (42.00 GiB) exceeds the size of thin pool pve/data and the amount of free space in volume group (<6.13 GiB).
Creating filesystem with 6553600 4k blocks and 1638400 inodes
Filesystem UUID: 2617e964-917a-4488-9cdc-41432282f124
Superblock backups stored on blocks:
32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
extracting archive '/var/lib/vz/template/cache/debian-12-standard_12.7-1_amd64.tar.zst'
Total bytes read: 521902080 (498MiB, 26MiB/s)
Detected container architecture: amd64
Creating SSH host key 'ssh_host_dsa_key' - this may take some time ...
done: SHA256:Srl80WByxeSu2seEvN9N92gJlB8s/qp5dyorX3+FAkk root@router
Creating SSH host key 'ssh_host_ed25519_key' - this may take some time ...
done: SHA256:AnAvwvTpjOZsd511MuIsNP53mU4bjKLZovWhD7DQ root@router
Creating SSH host key 'ssh_host_rsa_key' - this may take some time ...
done: SHA256:uwgQoZ+8VlhZns/spQ7/HFRTTbKTaRckCpuccvVR0 root@router
Creating SSH host key 'ssh_host_ecdsa_key' - this may take some time ...
done: SHA256:+12S/shekiZZ4E+2XQRGHu0VnwcyuHe2g9eCbu6L+Y root@router
TASK OK

```

Tareas | Registro del cluster

## Iniciamos la máquina

X PROXMOX Virtual Environment 8.2.2 | Buscar

Vista por servidor | Contenedor 100 (router) en el nodo pve | Ninguna etiqueta | Documentación | **Crear VM** | **Crear CT** | **root@pam**

- Centro de datos
  - pve
    - 100 (router) **✓**
    - 101 (LXC1)
    - localnetwork (pve)
    - local (pve)
    - local-lvm (pve)

Resumen | Consola | Recursos | Red | DNS | Opciones | Historial de tareas

root@router:~#

### 3. Configuración de la interfaz

Comprobamos que la máquina **router** está conectada a vmbr100, tal y como la hemos configurado, y procedemos a configurar la nueva IP.



Configuraremos la interfaz vmbr0 como DHCP para que tenga el acceso al exterior, y la vmbr1 de forma estática y sin ninguna puerta de red ya que sólo va a ser una red interna.

The screenshot shows the 'Red' configuration for container 100 (router). The table lists network interfaces: 'net1' (eth1, vmbr100, static, MAC BC:24:11:86..., IP 10.0.0.1/24), 'net0' (eth0, vmbr0, static, MAC BC:24:11:79..., IP dhcp), and 'vmbr0' (vmbr0, static, MAC ff:ff:ff:ff:ff:ff, IP 192.168.0.25/24). The 'vmbr0' row is highlighted with a red box.

The terminal window shows the output of the 'ip a' command. It lists several interfaces, including 'lo', 'eth0', 'eth1', and 'vmbr0'. The 'vmbr100' interface is highlighted with a red box. Its details are as follows:

```
root@pve:/var/lib/vz/template/cache# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast master vmbr0 state UP group default qlen 1000
    link/ether 08:00:27:66:68:41 brd ff:ff:ff:ff:ff:ff
3: vmbr0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether 08:00:27:66:68:41 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.25/24 brd 192.168.0.25 scope global vmbr0
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe66:6841/64 scope link
        valid_lft forever preferred_lft forever
4: vmbr100: <NO-CARRIER,BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UNKNOWN group default qlen 1000
    link/ether 02:9c:3b:eb:1b:fc brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.100/24 brd 10.0.0.100 scope global vmbr100
        valid_lft forever preferred_lft forever
    inet6 fe80::9c:3bff:feeb:1bfc/64 scope link
        valid_lft forever preferred_lft forever
root@pve:/var/lib/vz/template/cache#
```

**Recordar: elegir driver Virtio**



En nuestra máquina virtual comprobamos nuestro fichero **/etc/network/interfaces** (pues ya configué la red en la interfaz de Proxmox) y, posteriormente hacemos **un ip a** para ver si tiene asignada su IP.

Contenedor 100 (router) en el nodo pve Ninguna etiqueta

<span style="font-size: 1.5em;">█</span> Resumen <span style="font-size: 1.5em;">&gt;</span> Consola <span style="font-size: 1.5em;">█</span> Recursos <span style="font-size: 1.5em;">↔</span> Red <span style="font-size: 1.5em;">🕒</span> DNS <span style="font-size: 1.5em;">⚙️</span> Opciones <span style="font-size: 1.5em;">📝</span> Historial de tareas <span style="font-size: 1.5em;">💾</span> Respaldo	<pre style="font-family: monospace; margin: 0;">GNU nano 7.2 auto lo iface lo inet loopback  auto eth0 iface eth0 inet dhcp  auto net1 iface net1 inet static     address 10.0.0.1/24</pre>
---	---

```
root@router:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: eth0@if5: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether bc:24:11:79:98:15 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.168.0.29/24 brd 192.168.0.255 scope global dynamic eth0
        valid_lft 84477sec preferred_lft 84477sec
    inet6 fe80::be24:11ff:fe79:9815/64 scope link
        valid_lft forever preferred_lft forever
9: net1@if10: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether bc:24:11:20:8c:cf brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.0.0.1/24 scope global net1
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe20:8ccf/64 scope link
        valid_lft forever preferred_lft forever
root@router:~#
```

## 4. Creación de un contenedor LXC

Creamos ahora el contenedor LXC con el nombre **cliente**. Volvemos a pinchar en crear un nuevo CT y configuramos con los datos dados.

Crear: Contenedor LXC

General Plantilla Discos CPU Memoria Red DNS Confirmar

Key ↑	Value
cores	2
features	nesting=1
hostname	cliente
memory	2048
net0	name=eth0,bridge=vmbr100,firewall=1,ip=10.0.0.2/24,gw=10.0.0.1
nodename	pve
ostemplate	local:vztmpl/debian-12-standard_12.7-1_amd64.tar.zst
pool	
rootfs	local-lvm:20
ssh-public-keys	
swap	2048
unprivileged	1
vmid	102

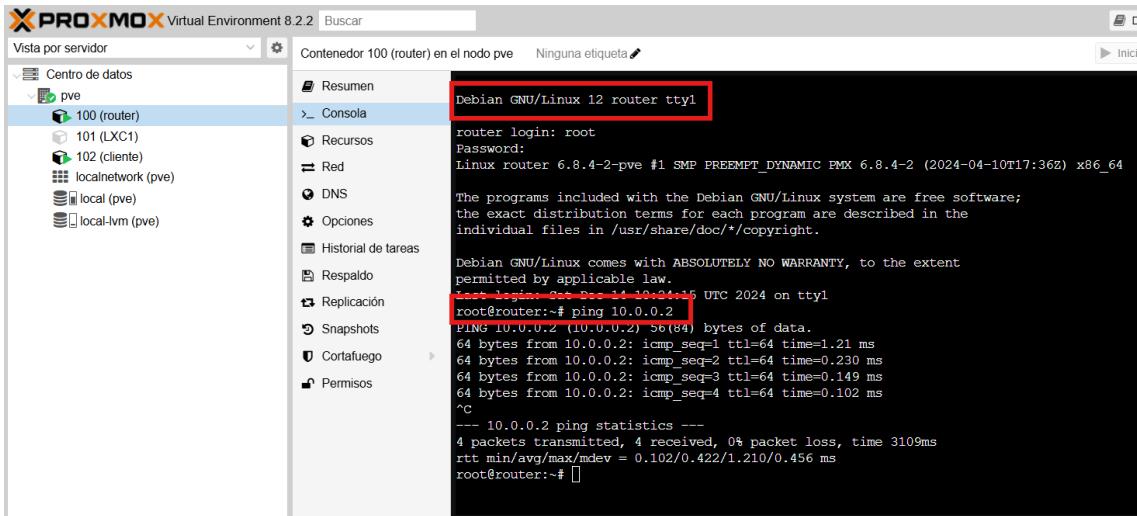
Iniciar después de la creación

Avanzado  Atrás Finalizar

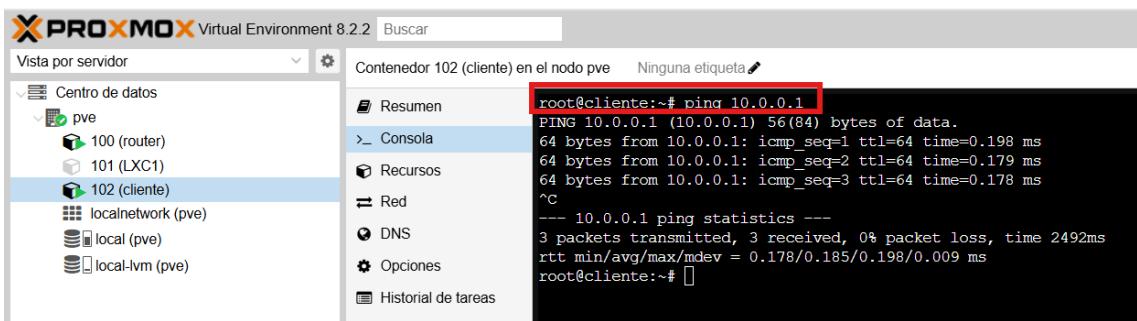
## 5. Comprobación conectividad

Iniciamos las máquinas **router** y **cliente**, y comprobamos la conectividad de cada una con respecto a la otra con un ping

Desde la VM router a la VM cliente



### Desde la VM cliente a la VM router



## 6. Instala en la máquina router un servidor web

Hacemos un apt update y posteriormente instalamos el servidor web apache2 con el comando apt install apache2 -y

```

root@router:~# apt install apache2 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data
  apache2-utils libapr1
  libaprutil1
  libaprutil1-dbd-sqlite3
  libaprutil1-lua

```

Iniciamos y verificamos que el servidor esté funcionando correctamente con systemctl start apache2 y systemctl enable apache2

pve - Proxmox Console - Google Chrome

No es seguro <https://192.168.0.25:8006/?console=lxc&xtermjs=1&vmid=100&vmname=...>

```
root@router:~# systemctl start apache2
root@router:~# systemctl enable apache2
Synchronizing state of apache2.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable apache2
root@router:~# 
```

Comprobamos el estado con systemctl status apache2 y verificamos el acceso local con https:// y la ip de la VM router.

pve - Proxmox Console - Google Chrome

No es seguro https://192.168.0.25:8006/?console=lx&xtermjs=1&vmid=100&vmname=...

```
root@router:~# systemctl status apache2
* apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; preset: enabled)
  Active: active (running) since Tue 2024-12-17 22:31:19 UTC; 4min 22s ago
    Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 1119 (apache2)
     Tasks: 55 (limit: 2284)
    Memory: 5.9M
      CPU: 206ms
     CGroup: /system.slice/apache2.service
             |-1119 /usr/sbin/apache2 -k start
             |-1120 /usr/sbin/apache2 -k start
             `-'1121 /usr/sbin/apache2 -k start

Dec 17 22:31:19 router systemd[1]: Starting apache2.service - The Apache HTTP Server...
Dec 17 22:31:19 router systemd[1]: Started apache2.service - The Apache HTTP Server.
root@router:~# 
```

**7. Activa el cortafuegos para la interfaz que está conectada a vmbr0.**

## En la interfaz de Proxmox:

Activamos el cortafuego globalmente Centro de Datos → Firewall → Opciones → Enable

Centro de datos	
Grupos	Cortafuego No
Conjuntos	Cortafuego Sí
Roles	Tasa Límite de registro Por defecto (enable=1,rate1/second,burst=5)
Dominios	Política de entrada DROP
HA	Política de salida ACCEPT
SDN	
Zonas	
VNets	
Opciones	
IPAM	
Cortafuego	
Grupo de seguridad	

**Editar: Cortafuego**  
 Cortafuego:

Observamos que se activa directamente en pve.

Opciones	
Cortafuego	Sí
Filtro SMI IREFS	Sí

En la VM router, tendremos que hacer lo mismo que en el global para activarlo.

Cortafuego	
Cortafuego	Sí
DHCP	Sí
NDP	Sí
Anuncio de enrutador	No
Filtro MAC	Sí
Filtro IP	No
log_level_in	nolog
log_level_out	nolog
Política de entrada	DROP
Política de salida	ACCEPT

Activamos el cortafuego en la interfaz específica:  
100(router) → Red → vmbr0 → Editar → marcar Cortafuegos.

## 8. Comprueba que no tenemos acceso al servidor web desde el exterior.

Compruebo el acceso y no puedo acceder

## 9. Crea una regla de cortafuego de entrada para que sea accesible el puerto 80 del protocolo TCP.

100(router) → Cortafuegos → Agregar y configuramos la regla

## 10. Comprueba que ya tenemos acceso al servidor web desde el exterior

```

root@router:~$ ping google.com
PING google.com (142.250.201.78) 56(84) bytes of data.
64 bytes from mad07s25-in-f14.le100.net (142.250.201.78): icmp_seq=1 ttl=117 time=24.3 ms
64 bytes from mad07s25-in-f14.le100.net (142.250.201.78): icmp_seq=4 ttl=117 time=101 ms
^C
--- google.com ping statistics ---
4 packets transmitted, 2 received, 50% packet loss, time 3275ms
rtt min/avg/max/mdev = 22.834/53.895/83.616/24.832 ms

root@router:~$ ping 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=57 time=55.2 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=57 time=22.8 ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=57 time=83.6 ms
^C
--- 1.1.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2120ms
rtt min/avg/max/mdev = 22.834/53.895/83.616/24.832 ms

```