



Redes en Proxmox II

SDN (Redes definidas por software) para MV y contenedores

- [Capítulos 1, 2 y 3 en el libro: Redes Proxmox I](#)
- [4.- SDN \(Software Defined Network\)](#)
 - [4.1.- ¿Qué son las SDN?](#)
 - [4.2.- SDN Simple. Una red SNAT de MV y contenedores](#)
 - [4.3.- SDN VLAN.](#)
 - [4.4.- SDN QinQ.](#)
 - [4.5.- SDN VxLAN.](#)
 - [4.6.- SDN EVPN.](#)
- [5.- Licencia y autoría de este material](#)



Capítulos 1, 2 y 3 en el libro: Redes Proxmox I

[Redes Proxmox I](#)



4.- SDN (Software Defined Network)

Redes definidas por software



4.- SDN (Software Defined Network)

4.1.- ¿Qué son las SDN?



Proxmox Server Solutions GmbH· Proxmox SDN (Todos los derechos reservados)

Las redes definidas por software son un paradigma que separa el plano de control del plano de datos en la arquitectura de red. A diferencia de las configuraciones de red tradicionales donde los planos de control y datos están estrechamente acoplados dentro de los dispositivos de red, SDN centraliza la lógica de control, lo que **permite una gestión de red dinámica y programable**. Este control centralizado permite un aprovisionamiento, gestión y optimización de la red más eficiente.

Proxmox VE, conocido por su flexibilidad y entorno rico en funciones, integra SDN para brindar a los usuarios capacidades de red avanzadas. Estos son los aspectos clave de SDN en Proxmox y su impacto en la infraestructura de virtualización:

- Integración de Open vSwitch:
 - Proxmox VE aprovecha Open vSwitch (OVS), un potente conmutador de software de código abierto, para implementar funcionalidades SDN. Open vSwitch actúa como un conmutador virtual dentro de Proxmox, proporcionando una forma flexible y programable de gestionar el tráfico de red entre máquinas virtuales y contenedores.
- Virtualización de red:
 - SDN en Proxmox permite la virtualización de red, lo que permite a los administradores crear redes aisladas y segmentadas lógicamente para diferentes



aplicaciones, proyectos o inquilinos. Esta capacidad mejora la seguridad, simplifica la gestión de la red y facilita la creación de topologías de red complejas dentro de un entorno virtualizado.

- Control de red centralizado:
 - con SDN, Proxmox centraliza el control de la red, proporcionando una vista unificada de toda la red. Este control centralizado simplifica la configuración, el monitoreo y la resolución de problemas de la red. Los administradores pueden ajustar dinámicamente las políticas y configuraciones de la red sin la necesidad de tocar dispositivos físicos individuales.
- Asignación dinámica de recursos:
 - SDN en Proxmox permite la asignación dinámica de recursos dentro de la red. Esto significa que los recursos de red se pueden aprovisionar y ajustar sobre la marcha, optimizando el uso del ancho de banda y garantizando que las aplicaciones reciban los recursos de red necesarios según la demanda.
- Aislamiento de tráfico y calidad de servicio (QoS):
 - SDN permite un control detallado sobre el tráfico de red, lo que permite el aislamiento de diferentes tipos de tráfico y la implementación de políticas de calidad de servicio (QoS). Esto es crucial para garantizar que las aplicaciones críticas reciban el ancho de banda y la prioridad necesarios, mejorando el rendimiento general de la red.

Nueva forma de visualización de las redes en Proxmox VE 8.1

Desde la versión de Proxmox VE 8.1, las redes (o zonas de redes), incluida la de por defecto "localnetwork", se visualizan como un recurso más en Proxmox VE, al nivel de las MV, contenedores, almacenamiento y Pool de usuarios:

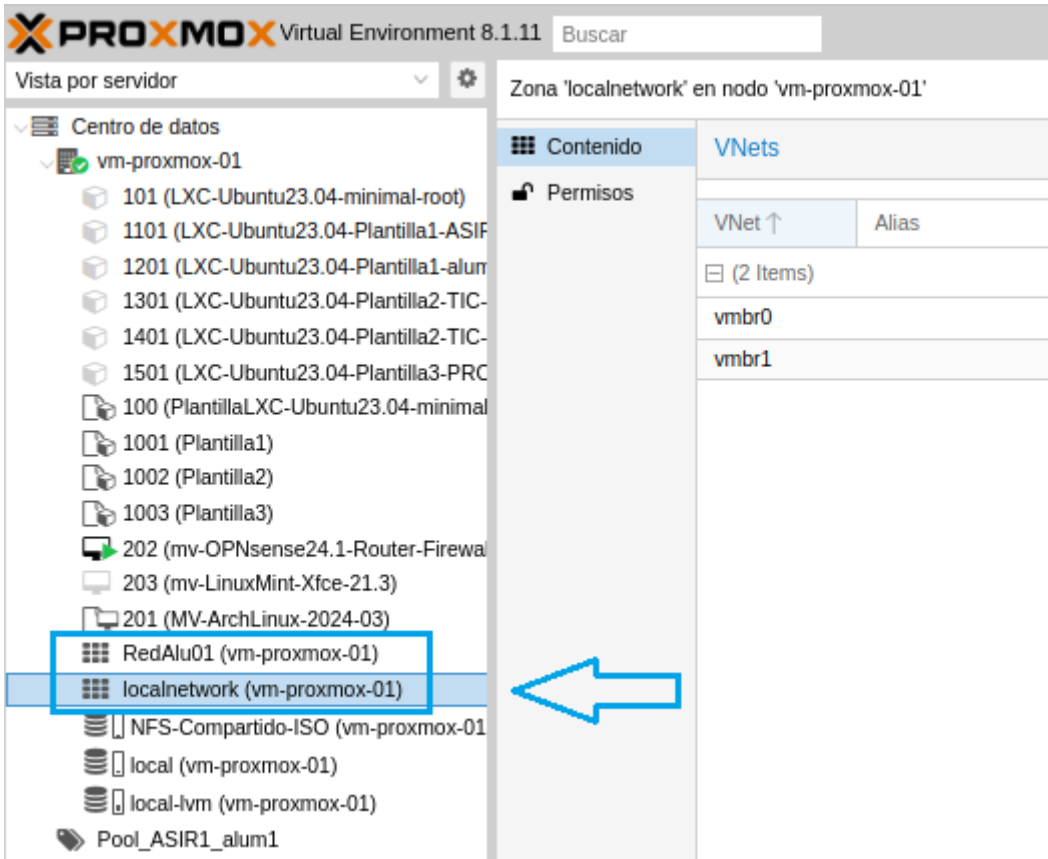


Imagen de elaboración propia: Recursos de Redes SDN en Proxmox VE 8.1 ([CC BY-NC-SA](https://creativecommons.org/licenses/by-nc-sa/4.0/))

Instalación

Si instalaste Proxmox sobre una instalación existente de Debian 12, deberás instalar manualmente algunos paquetes que no están en la lista predeterminada en la documentación de Proxmox:

```
apt install libpve-network-perl
```

También asegúrese de que su archivo `/etc/network/interfaces` contenga la línea:

```
source /etc/network/interfaces.d/*
```

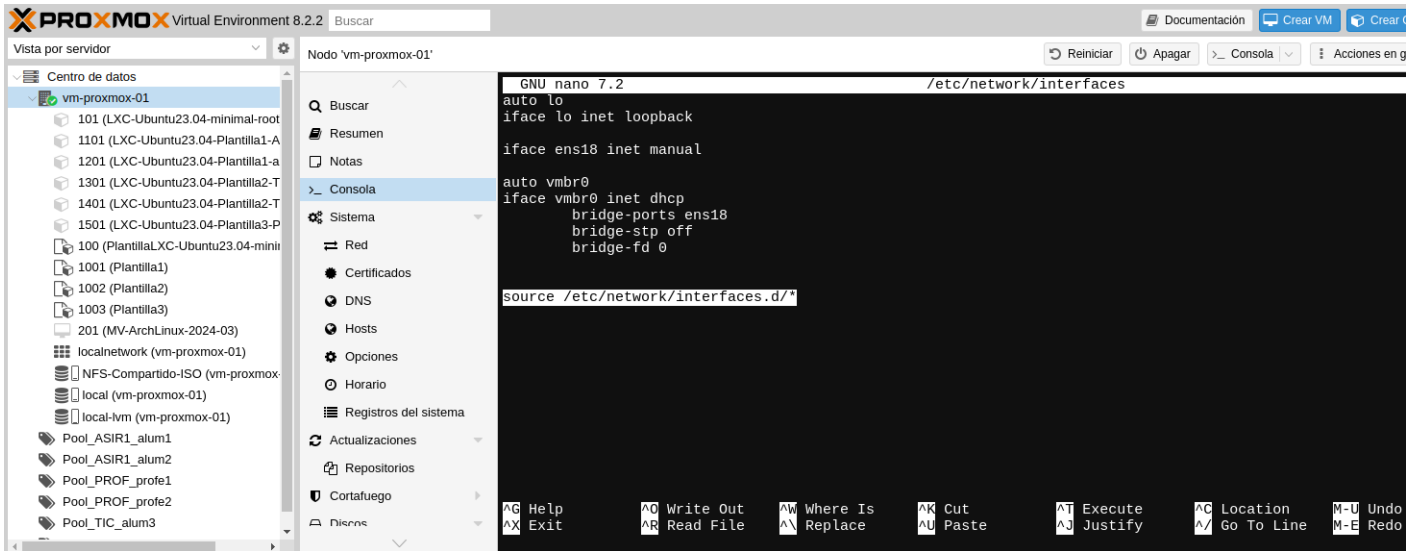


Imagen de elaboración propia: *Modificación del fichero /etc/network/interfaces del nodo de Proxmox* (CC BY-NC-SA)

También instalamos todo lo necesario para **el servidor de DHCP IPAM**, según la documentación oficial:

```
apt update
apt install dnsmasq
systemctl disable --now dnsmasq
```

Y para el enrutamiento se utiliza **FRRouting**, teniendo que instalar el siguiente paquete:

```
apt install frr-pythontools
```

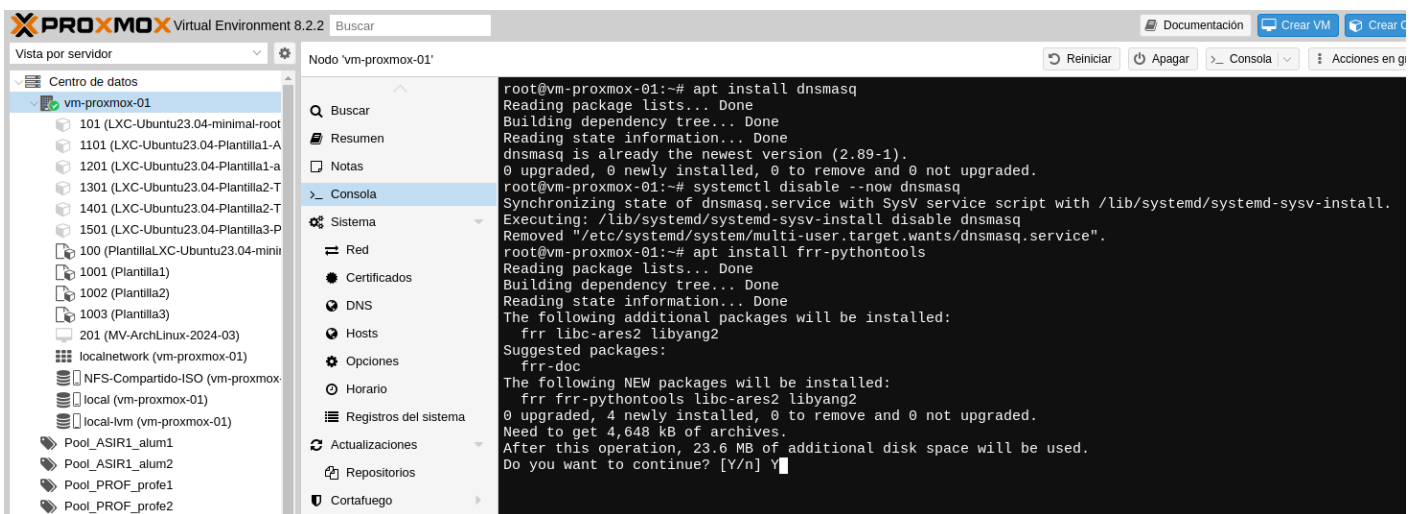


Imagen de elaboración propia: *Instalación de paquetes requeridos para la gestión de SDN* (CC BY-NC-SA)





4.- SDN (Software Defined Network)

4.2.- SDN Simple. Una red SNAT de MV y contenedores

1º. Empezar a crear redes SDN con un **Bridge-Networks-Simple** y le conectaremos dos contenedores, permitiéndoles comunicarse entre sí y al mismo tiempo usar DHCP para no configurar IP estáticas. La salida hacia el exterior se realizará mediante Source NAT.

En primer lugar, hay que asegurarse de que su archivo `/etc/network/interfaces` contenga la línea

```
source /etc/network/interfaces.d/*
```

También instalamos todo lo necesario para DHCP IPAM:

```
apt install dnsmasq  
systemctl disable --now dnsmasq
```

2º. **Las redes SDN se configuran a nivel del "Centro de datos"** para que en el caso de tener un cluster de nodos Proxmox, todos los nodos compartan la mismas redes y por tanto, puedan realizarse correctamente las comunicaciones, replicas, migraciones de las MV y contenedores entre los nodos Proxmox que conformen el cluster.

Crear la Zona

A continuación, navegaremos a la pestaña Zonas, hacer clic en "Agregar" y luego seleccionar "Simple".

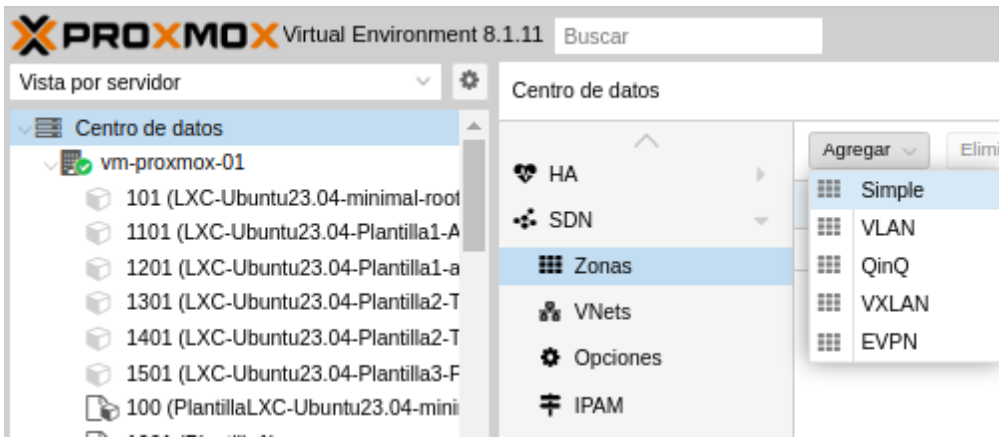


Imagen de elaboración propia: *Crear una Zona SDN del tipo "Simple"* ([CC BY-NC-SA](#))

Una Zona en SDN de Proxmox es un conjunto de VNets (redes virtuales) que posteriormente podremos enrutar si queremos. Podemos hacer el símil entre Zona y una red privada (LAN o MAN). Mientras que una VNet sería como un dominio de broadcast, podemos ver a las **VNet como switch** con un puente hacia la raíz de su Zona. Por tanto, podremos elegir si queremos que MV o contenedores en una VNet puedan o no comunicarse con otras MV y contenedores de otra VNet de su misma Zona.

Asegúrese de darle a su red un nombre descriptivo (ID) y asegúrese de habilitar DHCP automático expandiendo las opciones avanzadas:

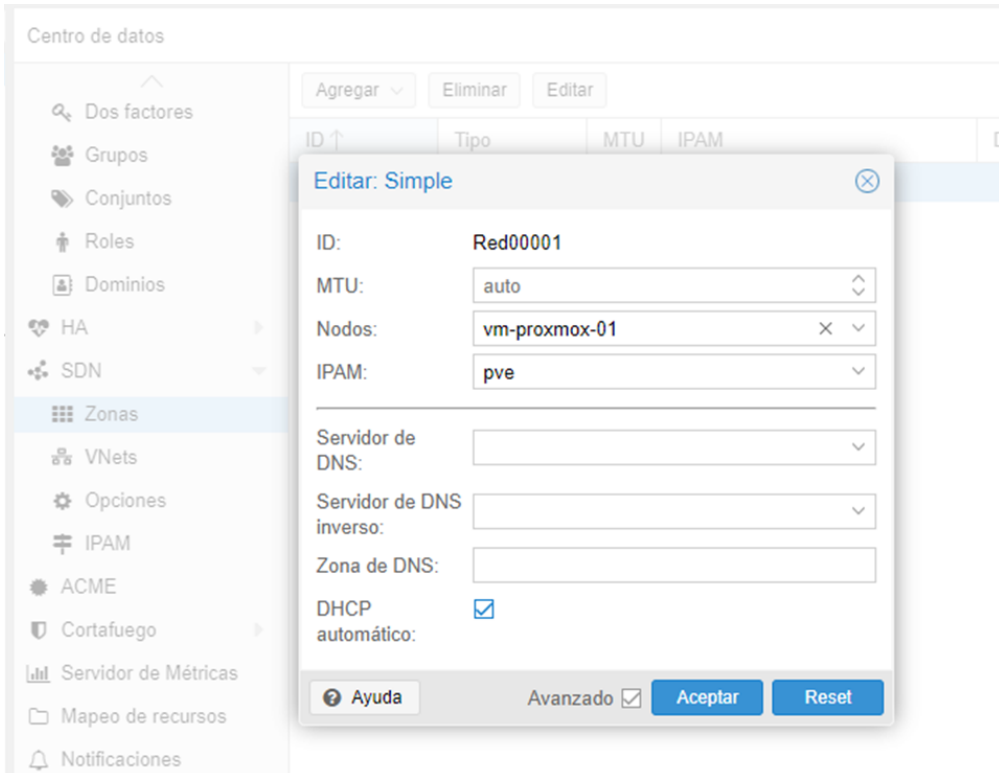


Imagen de elaboración propia: *Crear/Editar una Zona nueva del tipo Simple llamada "Red00001"* ([CC BY-NC-SA](#))



Cada vez que hagamos un cambio en una SDN tendremos que reiniciar los cambios para que estos surjan efecto, también podemos dejar el reinicio del servicio para el final de la configuración y hacerlo una sola vez. Para hacer el reinicio del servicio de red tendremos que ir a SDN --> "Aplicar":

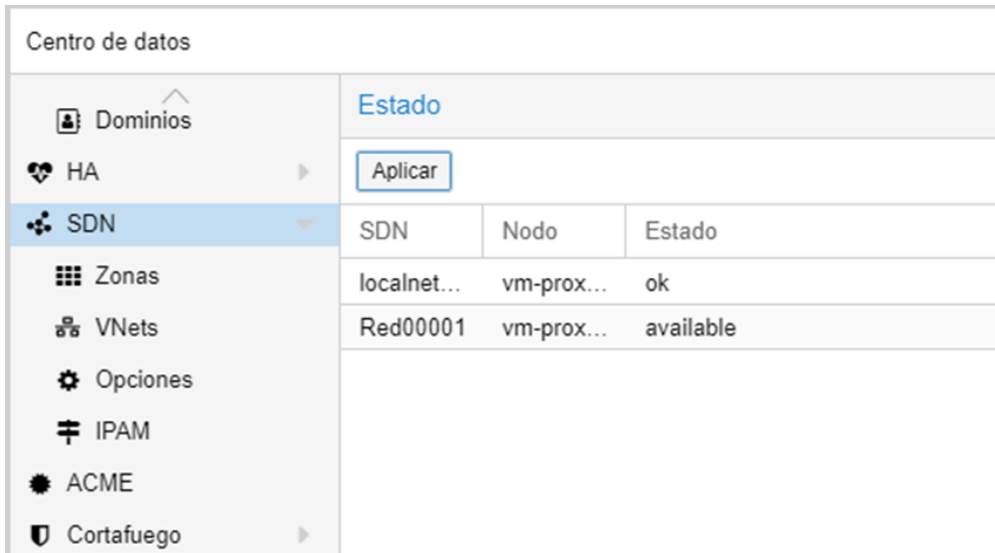


Imagen de elaboración propia: *Aplicar cambios en la configuración de las SDN* (CC BY-NC-SA)

Y ya podremos ver el nuevo recurso de Zona en el nodo de Proxmox:

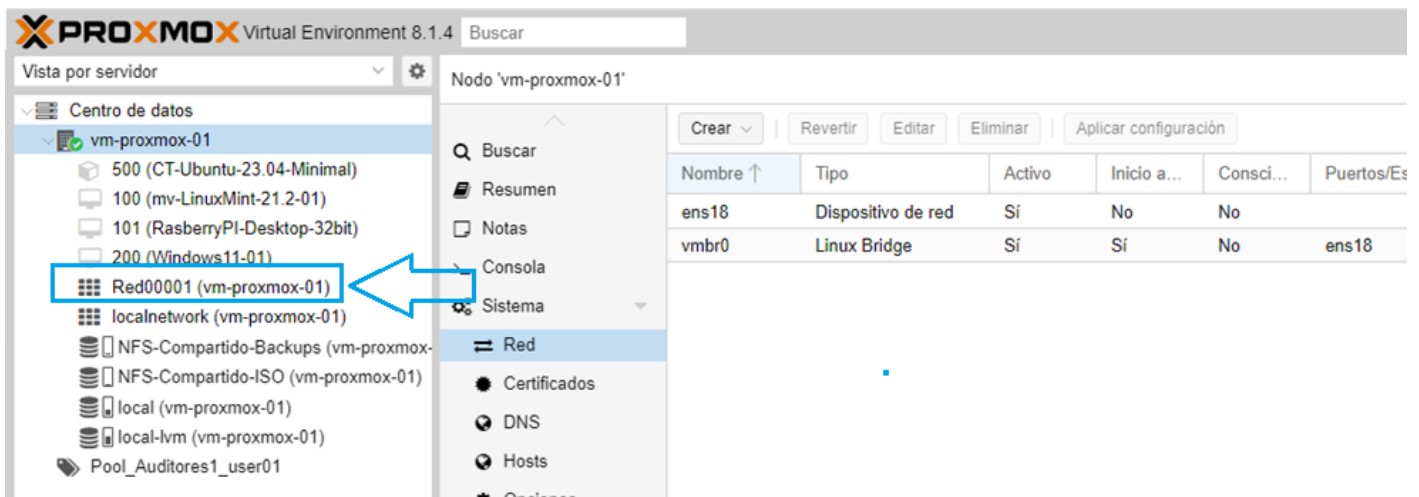


Imagen de elaboración propia: *Nueva Zona SDN creada* (CC BY-NC-SA)

3º. Después de eso necesitamos crear una nueva VNet, que es básicamente un switch virtual con puente hacia la raíz de su Zona:

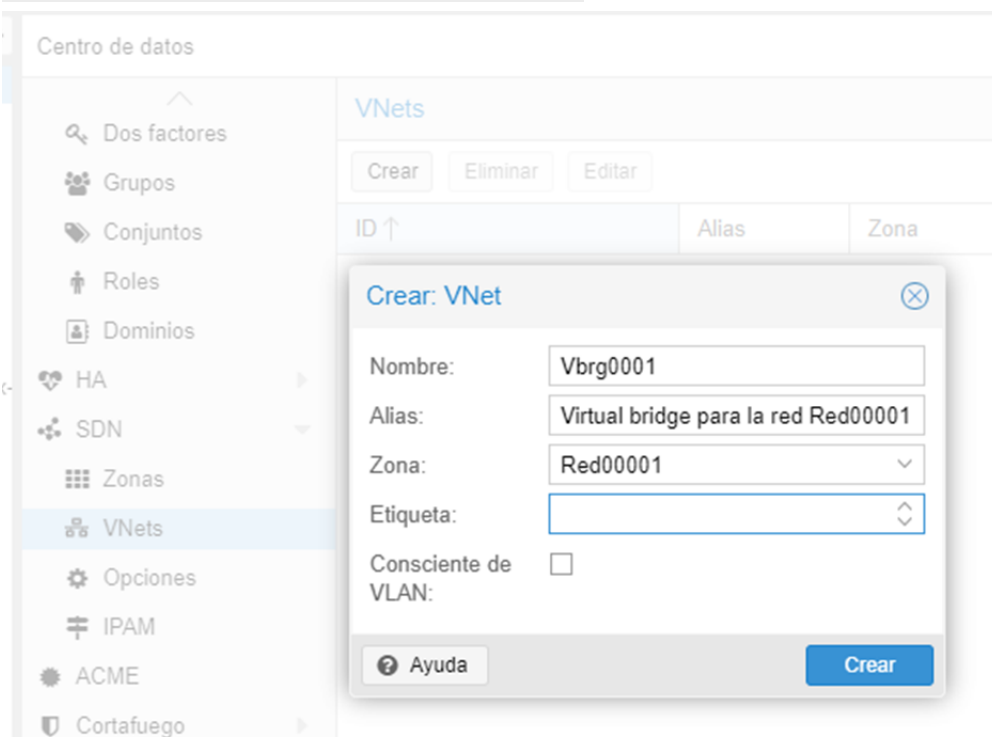


Imagen de elaboración propia: Creación de una VNet llamada "Vbrg0001", dentro de la Zona "Red00001" (CC BY-NC-SA)

4º. Como queremos usar DHCP y no configurar manualmente las interfaces de red en nuestros contenedores y/o MV, agregamos una "subred", que también contiene información sobre los rangos de DHCP que queremos ofrecer a los clientes. En realidad, aunque en Proxmox nos permita realizar tantas subredes como queramos dentro de una misma VNet, este componente (versión de Proxmox VE 8.2 en el momento de realizar este manual) solo nos sirve para configurar el direccionamiento del servidor DHCP de la propia SDN, porque no podemos conectar directamente una subred a una interfaz de red de una MV o de un contenedor, sino que eso se hace a nivel de VNet (en nuestro ejemplo "Vbrg0001").

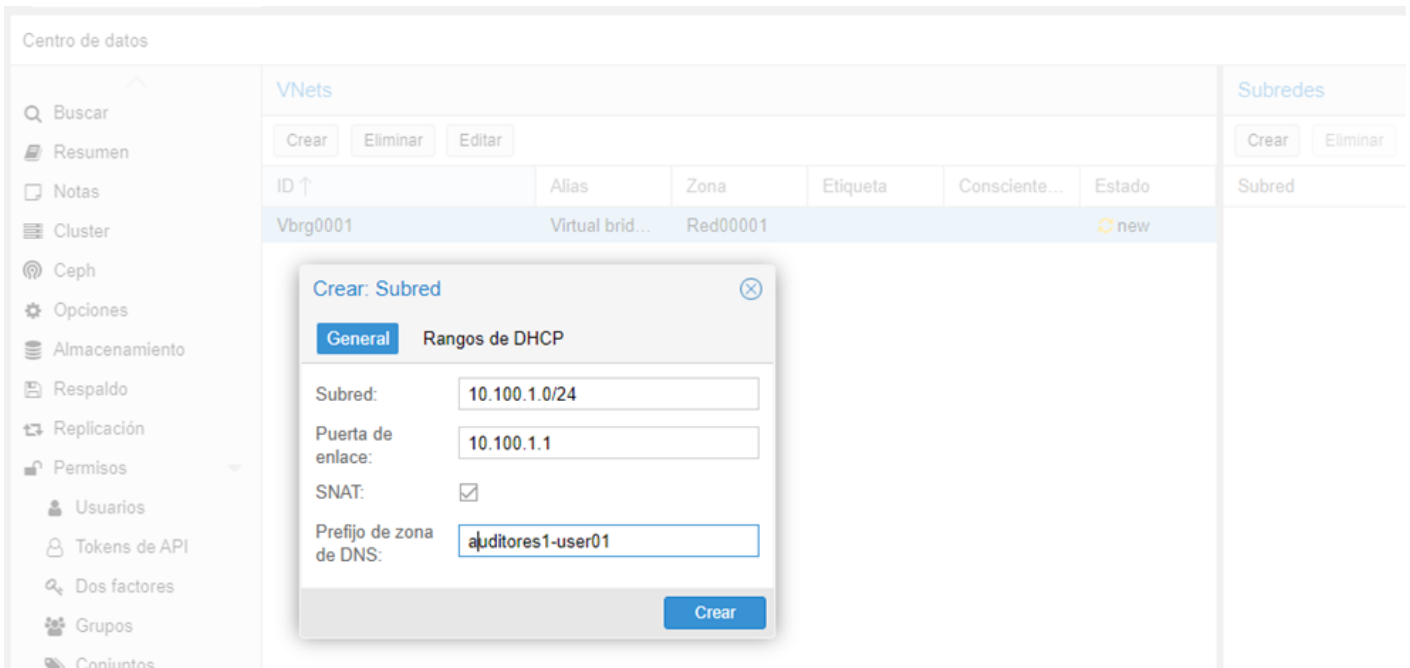


Imagen de elaboración propia: Creación de la subred 10.100.1.0/24 (CC BY-NC-SA)

¡ATENCIÓN! La dirección de la puerta de enlace es una dirección que nosotros elegiremos entre las posibles direcciones IP de la subred que estamos creando. Proxmox creará un servicio Source NAT en esa IP, hacia la VNet, que a su vez hará de puente con la raíz de su Zona y de hay otro puente hacia la interfaz de red del nodo Proxmox con salida a su WAN. **Es necesario que el servicio SNAT esté habilitado** (chequeado en la subred).

Debemos ahora asignar el rango de IP que serán concedidas por DHCP dentro la subred:

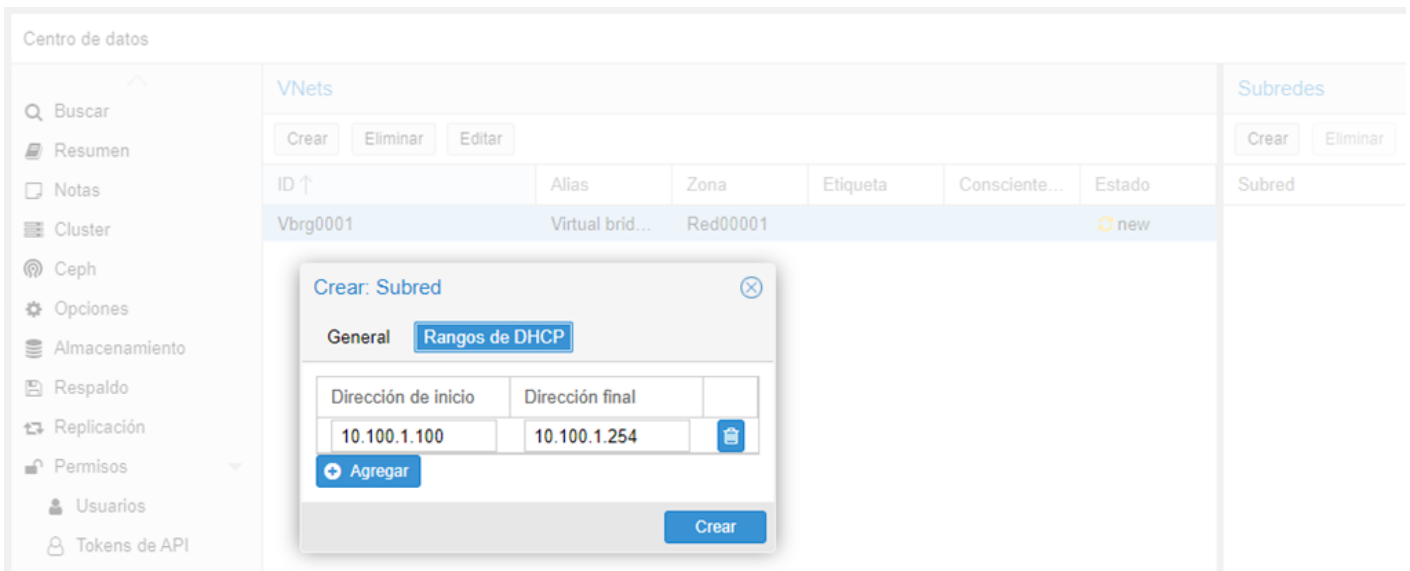


Imagen de elaboración propia: Rango de DHCP dentro de la subred 10.100.1.0/24 (CC BY-NC-SA)



Como puedes ver, la configuración inicial es sencilla. Todo lo que acabamos de crear tiene el estado "nuevo", lo que significa que aún no se ha aplicado. Para aplicar cambios, necesitamos recargar la red, lo que podemos hacer con el botón "Aplicar" en la página principal de descripción general de SDN.

Si estamos utilizando permisos para los usuarios y/o grupos de Proxmox, no se nos debe olvidar conceder permisos para poder utilizar la Zona, con todos sus VNet (switch virtuales), o solo a una VNet en particular:

Centro de datos		Ayuda		
Buscar Resumen Notas Cluster Ceph Opciones	Agregar	Eliminar		
	Ruta ↑	Usuario/Grupo/Token d...	Rol	Propagar
	/pool/Pool_Audidores1_user01	Audidores1_user01@pve	PVEAdmin	true
	/sdn/zones/Red00001	Audidores1_user01@pve	PVESDNUser	true
	/vms	@grupoAudidores1	PVEAuditor	true

Imagen de elaboración propia: Conceder permisos a una Zona para que pueda ser utilizada por un usuario en particular (CC BY-NC-SA)

En estas versiones nuevas de Proxmox puede surgir un error después de crear o editar una SDN. Si te surge el "ERROR 400: poolid: property is not defined in schema and the schema does not allow additional properties", se soluciona recargando el navegador web con **F5** del GUI de Proxmox:

The screenshot shows the 'Agregar: Permisos de usuario' dialog box in Proxmox. The fields are filled with: Ruta: /sdn/zones/Red00001, Usuario: Audidores1_user01@pve, Rol: PVESDNUser, and Propagar: checked. An error dialog box is overlaid on top, displaying the message: 'Error: Parameter verification failed. (400) poolid: property is not defined in schema and the schema does not allow additional properties'. There is an 'Aceptar' button at the bottom of the error dialog.



Si queremos crear una red aislada de MV y/o contenedores, solo tenemos que **deshabilitar SNAT de la subred** y no configurar la puerta de enlace virtual.

Si por el contrario, queremos tener una MV o contenedor proporcionando un servicio a la WAN, tendremos que abrir los puertos necesarios utilizando reglas del cortafuegos de Proxmox (ver capítulo 4 cortafuegos).

5. Comprobaremos el funcionamiento de la VNet utilizando una MV:

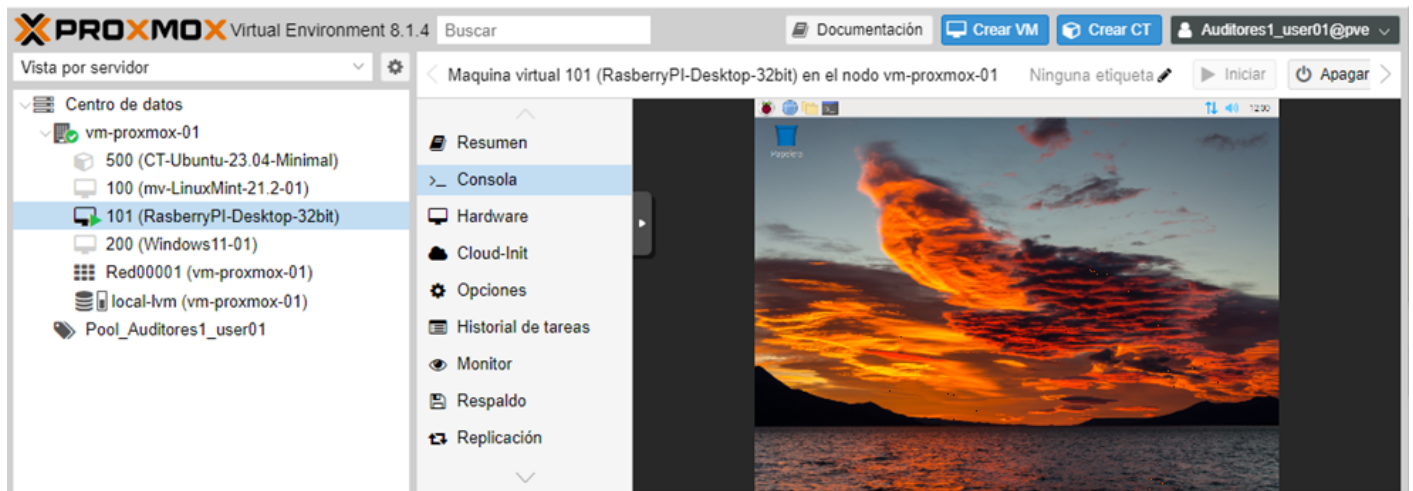


Imagen de elaboración propia: *MV 101 con RaspberryPI* ([CC BY-NC-SA](#))

Editaremos su interfaz de red para conectarla al switch virtual de la VNet "Vbrg0001":

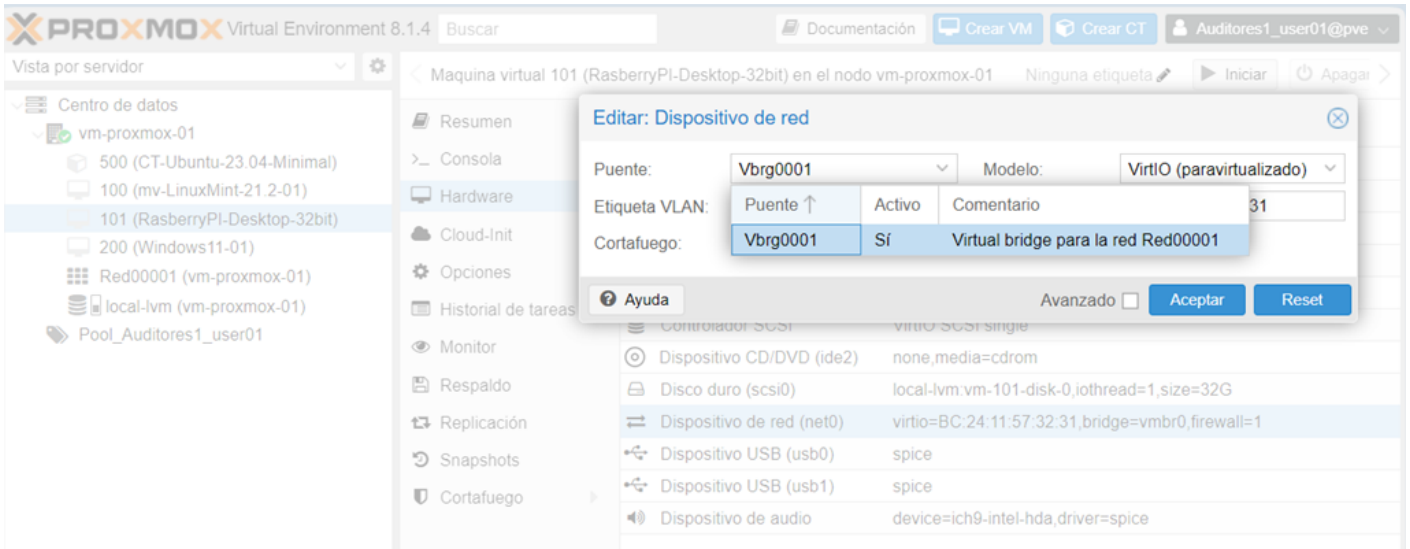


Imagen de elaboración propia: *Editar la interfaz de red de la MV* (CC BY-NC-SA)

Y comprobaremos la asignación de la configuración de red por DHCP y comprobaremos la conexión a los DNS de Google:

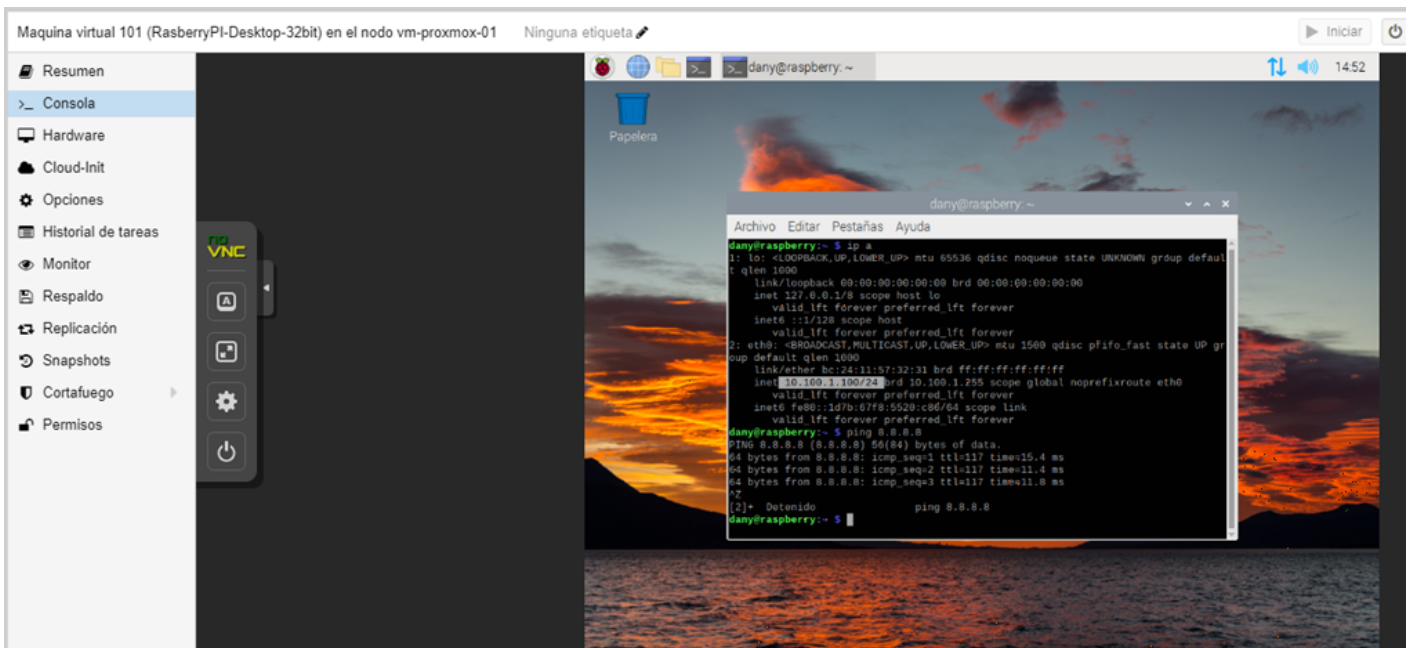


Imagen de elaboración propia: *Comprobar la IP y ping a 8.8.8.8* (CC BY-NC-SA)

6º. Finalmente podemos observar el mapa de asignación de IP del servicio de DHCP de Proxmox:



Centro de datos

Recargar

Nombre / VMID ↑	Dirección IP ↑	MAC	Puerta de enlace	Acciones
Red00001				
Vbrg0001				
10.100.1.0/24				
Puerta de enlace	10.100.1.1		1	
101	10.100.1.100	BC:24:11:57:32:31		 

Imagen de elaboración propia: Consulta de la IP asignadas por DHCP IPAM ([CC BY-NC-SA](#))

Para saber más

Consulta la [documentación oficial de Proxmox](#)



4.- SDN (Software Defined Network)

4.3.- SDN VLAN.

¿Qué es una VLAN?

Dirección Origen	Dirección Destino	Tipo de trama	Datos (46-1500)	Frame Check Sequence (FCS/CRC)
------------------	-------------------	---------------	-----------------	--------------------------------

Trama Ethernet

[Eduardo Taboada \(Tecnocratica.net\)](#) · *Trama Ethernet. Capa 2 de TCP/IP* (Todos los derechos reservados)

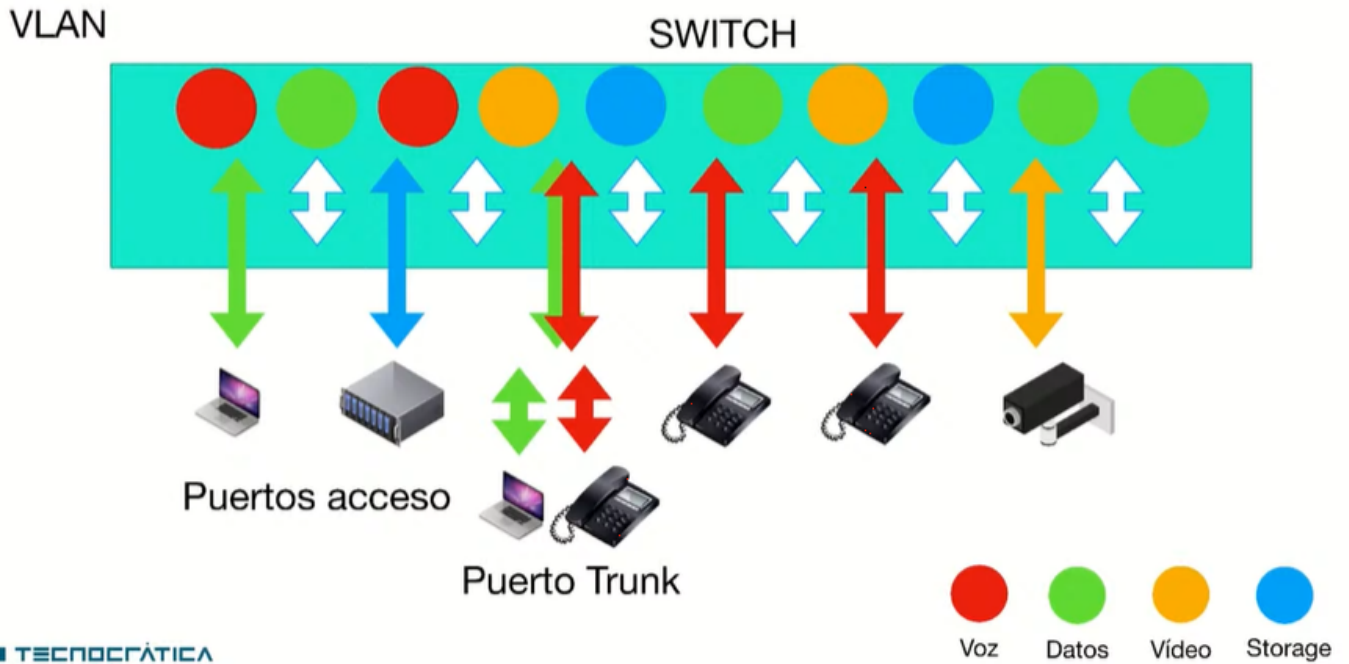
VLAN o también conocidas como **redes de área local virtuales**, es una tecnología de redes que nos permite crear redes lógicas independientes dentro de la misma red física, es decir, sobre una mismo dominio de broadcast (LAN) segmentaremos la red física en redes virtuales y esto lo lograremos poniendo una "Tag" etiqueta dentro de la trama Ethernet indicando el número de VLAN asignada (0 a 4096, es decir 12 bits)

Las VLAN nos permite crear redes lógicamente independientes, por tanto, podemos aislarlas para que solamente tengan conexión a Internet, y denegar el tráfico de una VLAN a otra. Por defecto no se permite a las VLANs intercambiar tráfico con otra VLAN, es totalmente necesario ascender a nivel de red (Capa 3 de TCP/IP) con un router o un switch multicapa, con el objetivo de activar el inter-vlan routing, es decir, el enrutamiento entre VLANs para sí permitir la comunicación entre ellas siempre que lo necesitemos.

Tengamos en cuenta que la Capa 2 de Ethernet está preparada para detectar colisiones en el medio de transmisión y para ello necesita una longitud de trama de datos (o MTU) mínima de 46 bytes y un máximo de 1500 más los encabezados y el CRC para que sean compatibles con la mayoría de los switch comerciales.

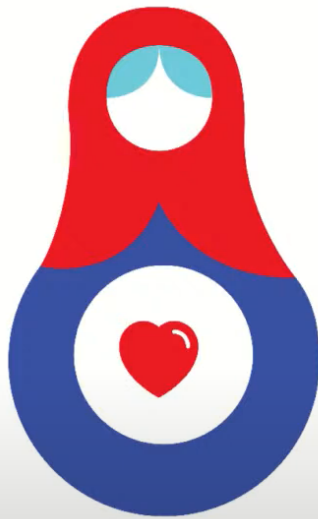


Es evidente que cuanto mayor sea el campo de datos de las tramas (MTU) mayor será el rendimiento pero cuidado con superar el MTU de 1500, porque muchos fabricantes de switch comerciales no lo soportan.



Eduardo Taboada (Tecnocratica.net) · Puertos de acceso y puertos Trunk para una VLAN (Todos los derechos reservados)

Para conectar un interfaz de red a una VLAN lo realizaremos a través de los "puertos de acceso" para ello debemos configurar la interfaz de red con el número de "Tag" de la VLAN que le corresponda y así el switch sabrá dividir el tráfico de red entre las distintas VLAN. Cuando en un mismo enlace queremos conectar varias VLAN del switch al host, necesitaremos hacerlo mediante un "puerto Trunk", para ello debemos habilitar en el switch el soporte a varias VLAN por el mismo puerto de salida.



Dirección Origen	Dirección Destino	Tipo de trama	Datos (46-1500)	Frame Check Sequence (FCS/CRC)
------------------	-------------------	---------------	-----------------	--------------------------------

Trama Ethernet

Dirección Origen	Dirección Destino	Tag	Tipo de trama	Datos (46-1500)	Frame Check Sequence (FCS/CRC)
------------------	-------------------	-----	---------------	-----------------	--------------------------------

Trama Ethernet con VLAN (voz)

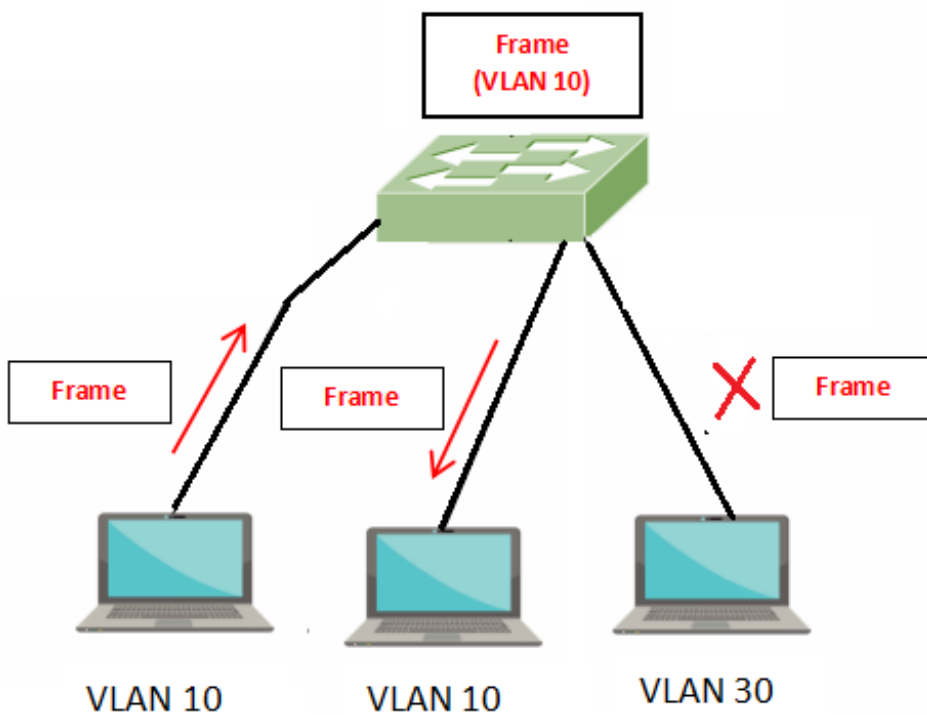


TECNOCRÁTICA

Eduardo Taboada (Tecnocratica.net) Esquema de trama Ethernet VLAN (Todos los derechos reservados)

Si queremos tener plena compatibilidad con todos los switch comerciales debemos reducir el tamaño MTU de 1500 para no tener problemas al haber agregado el "Tag" de la VLAN.

VLAN





Implementación de una SDN VLAN entre las MV y contenedores de dos nodos Proxmox

Cuando las máquinas virtuales de diferentes nodos necesitan comunicarse a través de una **red aislada**, la zona VLAN permite el aislamiento a nivel de red mediante etiquetas VLAN.

1º. Crea una Zona VLAN (recordar que las SDN se configuran a nivel de Centro de datos:

SDN	Nodo	Estado
localnet...	vm-prox...	ok
ZonaVL...	vm-prox...	pending

Hora de inicio	Hora final	Nodo	Nombre de usuario	Descripción	Estado
May 09 12:52:36	May 09 12:52:42	vm-proxm...	root@pam	SRV networking - Recargar	OK
May 09 12:52:33	May 09 12:52:42	vm-proxm...	root@pam	reloadnetworkall	OK

Imagen de elaboración propia: *Crear una VLAN* ([CC BY-NC-SA](#))

Llamaremos a la Zona como "ZonaVLAN" y la conectaremos al Bridge "vibr0":



Agregar: VLAN

ID:

Bridge:

MTU:

Nodos:

IPAM:

Servidor de DNS:

Servidor de DNS inverso:

Zona de DNS:

Avanzado

Imagen de elaboración propia: Creación de la Zona SDN del tipo VLAN (CC BY-NC-SA)

Crear una VNet denominada "SwVLAN10" con la etiqueta VLAN 10 en la Zona creada anteriormente:

Centro de datos

Permisos

- Usuarios
- Tokens de API
- Dos factores
- Grupos
- Conjuntos
- Roles
- Dominios
- HA
- SDN
- Zonas
- VNETs**
- Opciones
- IPAM
- ACME

VNETs

ID ↑	Alias	Zona	Etiqueta	Consci...	Estado
------	-------	------	----------	-----------	--------

Crear: VNet

Nombre:

Alias:

Zona:

Etiqueta:

Consciente de VLAN:

Imagen de elaboración propia: Creación de la VNet "SwVLAN10" (CC BY-NC-SA)

Aplica la configuración a través del panel principal de SDN:

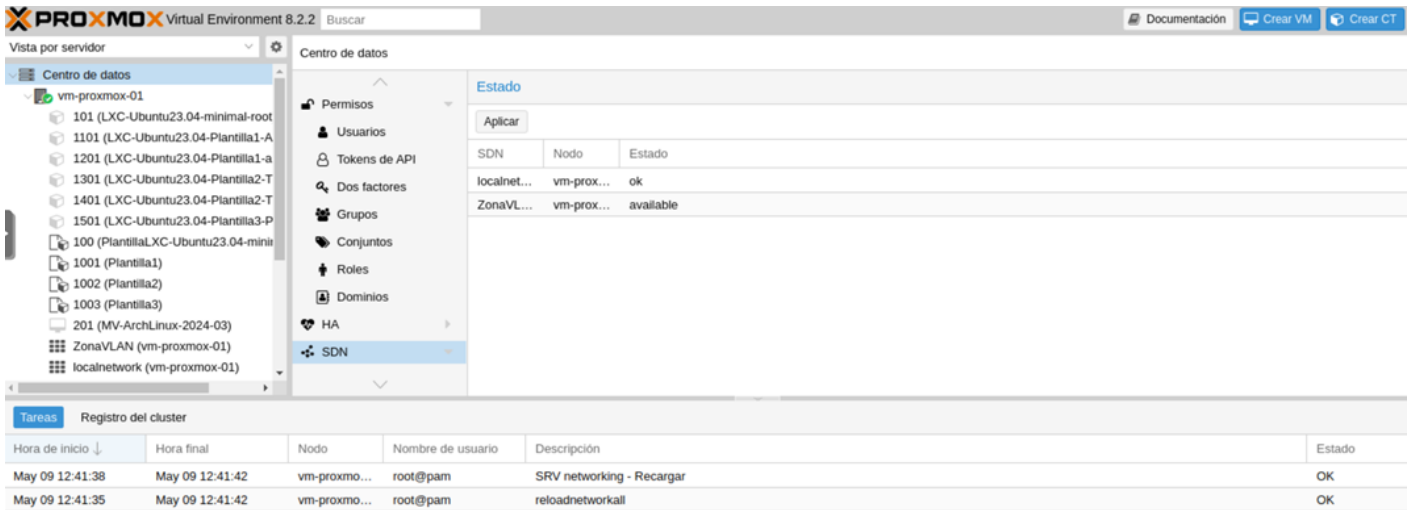


Imagen de elaboración propia: *Aplicar cambios en las SDN* (CC BY-NC-SA)

Crearemos otra VNet llamada "SwVLAN30" para comprobar que el tráfico de red se encuentra aislado entre distintas VLAN:

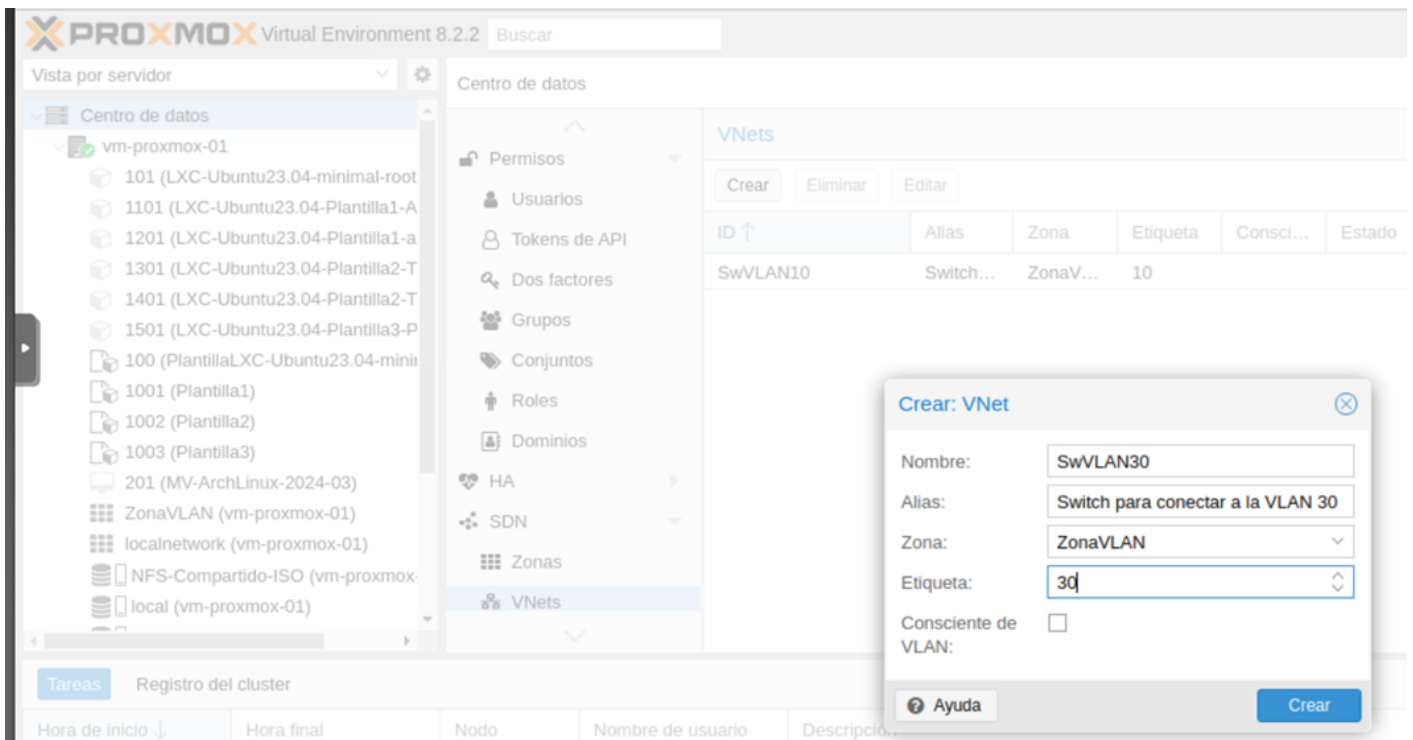


Imagen de elaboración propia: *Creación de otro Switch virtual para hacer otra VLAN distinta a la anterior* (CC BY-NC-SA)

Aplicamos nuevamente los cambios en la SDN.

Repetiremos el proceso en otro nodo Proxmox:

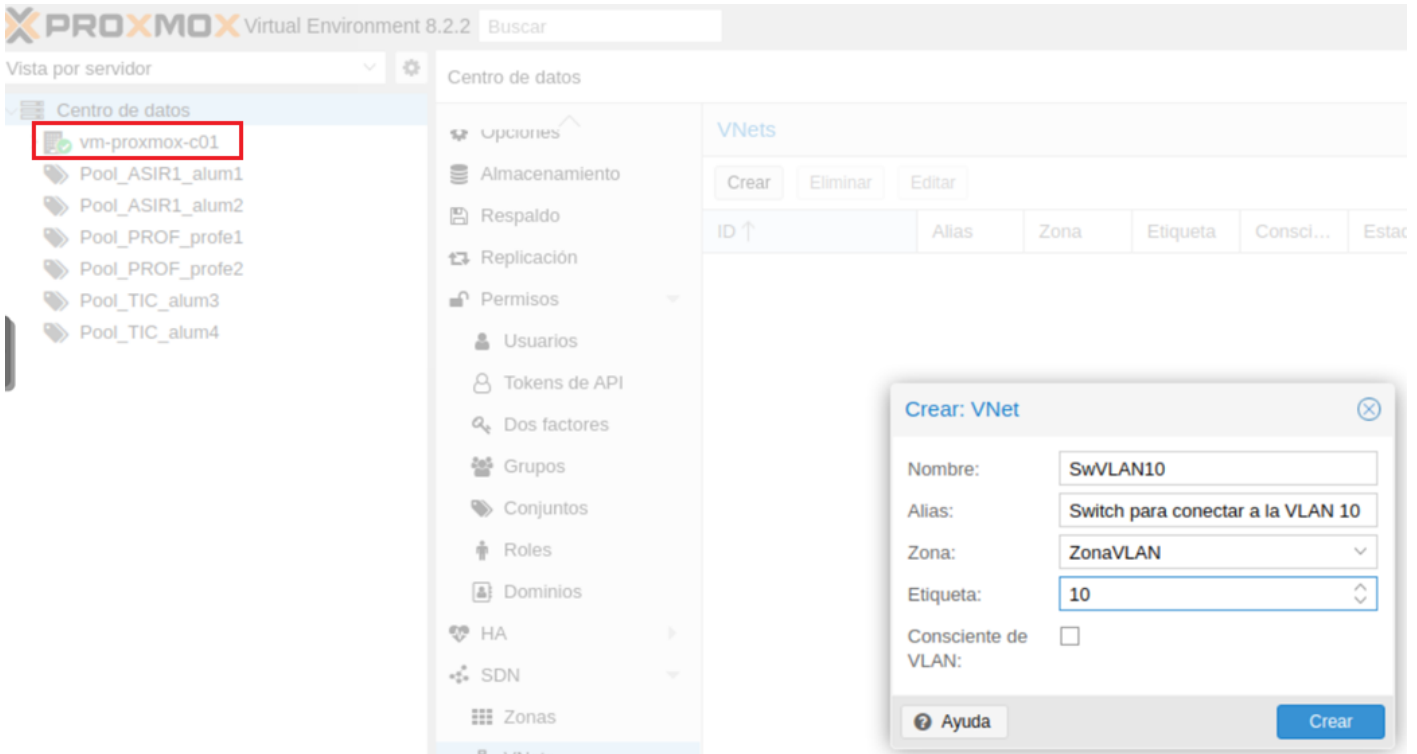


Imagen de elaboración propia: Repetir el proceso de creación de la Zona VLAN y VNet en otro nodo de Proxmox distinto al anterior ([CC BY-NC-SA](#))

Aplicamos cambios en el SDN de nuevo nodo Proxmox:

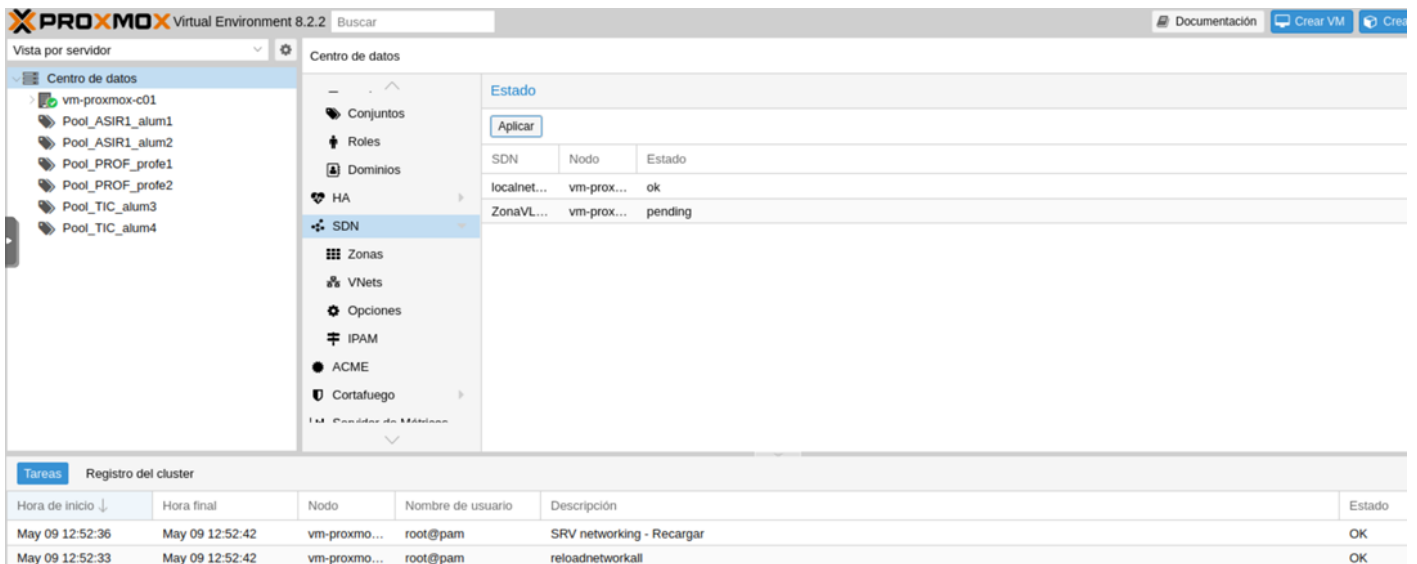


Imagen de elaboración propia: Aplicar cambios en el SDN ([CC BY-NC-SA](#))



Creamos contenedores para hacer las pruebas y le asignamos la IP de forma estática, ya que Proxmox no nos facilita un servidor DHCP para Zonas del tipo VLAN. Configuraremos un contenedor en cada nodo de Proxmox y asociaremos su interfaz de red a la VNet "SwVLAN10":

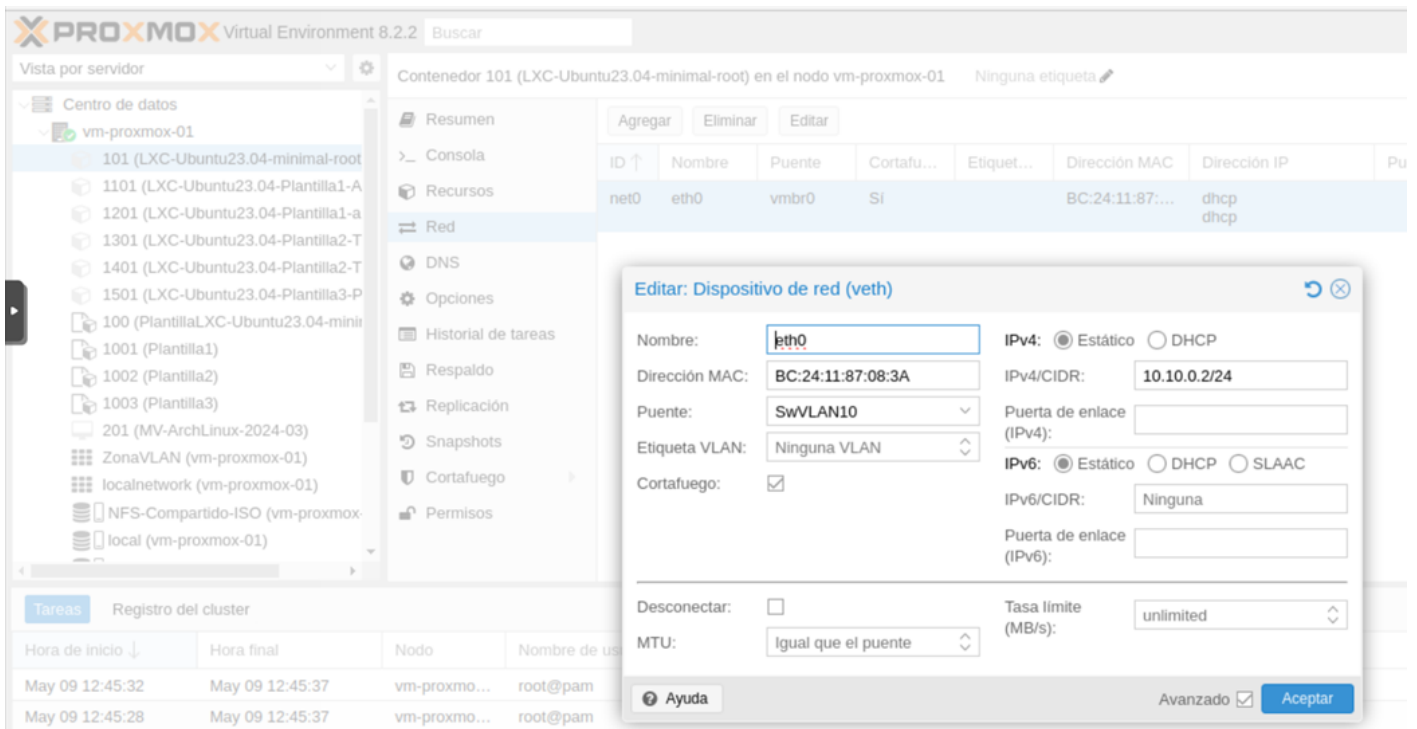


Imagen de elaboración propia: *Asignación de IP estáticas y conexión a la VNet "SwVLAN" de los contenedores* (CC BY-NC-SA)

Haremos otro contenedor y le asociaremos su interfaz de red al "SwVLAN30"

Comprobaremos el resultado haciendo ping entre los contenedores de la mismo VLAN 10:



```
root@LXC-Ubuntu23:~# ping 10.10.0.2
PING 10.10.0.2 (10.10.0.2) 56(84) bytes of data:
64 bytes from 10.10.0.2: icmp_seq=1 ttl=64 time=3.07 ms
64 bytes from 10.10.0.2: icmp_seq=2 ttl=64 time=1.05 ms
64 bytes from 10.10.0.2: icmp_seq=3 ttl=64 time=0.964 ms
64 bytes from 10.10.0.2: icmp_seq=4 ttl=64 time=1.13 ms
64 bytes from 10.10.0.2: icmp_seq=5 ttl=64 time=0.987 ms
^C
--- 10.10.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 0.964/1.439/3.066/0.815 ms
root@LXC-Ubuntu23:~#
```

```
PING 10.10.0.22 (10.10.0.22) 56(84) bytes of data:
64 bytes from 10.10.0.22: icmp_seq=1 ttl=64 time=5.90 ms
64 bytes from 10.10.0.22: icmp_seq=2 ttl=64 time=1.11 ms
64 bytes from 10.10.0.22: icmp_seq=3 ttl=64 time=1.10 ms
64 bytes from 10.10.0.22: icmp_seq=4 ttl=64 time=1.11 ms
64 bytes from 10.10.0.22: icmp_seq=5 ttl=64 time=1.04 ms
64 bytes from 10.10.0.22: icmp_seq=6 ttl=64 time=1.14 ms
^C
--- 10.10.0.22 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5008ms
rtt min/avg/max/mdev = 1.035/1.898/5.900/1.789 ms
root@LXC-Ubuntu23-VLAN10:~#
```

Imagen de elaboración propia: *Comprobación de la interconexión de 2 contenedores en la misma VLAN pero en distintos nodos Proxmox* (CC BY-NC-SA)

Ahora configuraremos la IP estática del tercer contenedor pero estará conectado a la VLAN 30:



Proxmox Virtual Environment 8.2.2

Vista por servidor: Centro de datos > vm-proxmox-01

Contenedor 1602 (LXC-Ubuntu23-VLAN30) en el nodo vm-proxmox-01

ID	Nombre	Puente	Cortafue...	Etiquet...	Dirección MAC	Dirección IP
net0	eth0	SwVLAN...	Si		BC:24:11:1B:...	10.10.0.2/24

Editar: Dispositivo de red (veth)

Nombre: IPv4: Estático DHCP

Dirección MAC: IPv4/CIDR:

Puente: Puerta de enlace (IPv4):

Etiqueta VLAN: IPv6: Estático DHCP SLAAC

Cortafuego: IPv6/CIDR:

Puerta de enlace (IPv6):

Desconectar: Tasa límite (MB/s):

MTU:

Avanzado

Imagen de elaboración propia: Configuración del tercer contenedor conectado a la VLAN30 (CC BY-NC-SA)

Podemos observar como el contenedor en la VLAN30 se encuentra aislado del tráfico de red de los contenedores que se encuentran el VLAN10:

Proxmox Virtual Environment 8.2.2

Contenedor 101 (LXC-Ubuntu23.04-NodoC01-01) en el nodo vm-proxmox-c01

```

root@LXC-Ubuntu23:~# ping 10.10.0.2
PING 10.10.0.2 (10.10.0.2) 56(84) bytes of data.
64 bytes from 10.10.0.2: icmp_seq=1 ttl=64 time=3.07 ms
64 bytes from 10.10.0.2: icmp_seq=2 ttl=64 time=1.05 ms
64 bytes from 10.10.0.2: icmp_seq=3 ttl=64 time=0.964 ms
64 bytes from 10.10.0.2: icmp_seq=4 ttl=64 time=1.13 ms
64 bytes from 10.10.0.2: icmp_seq=5 ttl=64 time=0.987 ms
^C
--- 10.10.0.2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 0.964/1.439/3.066/0.815 ms
root@LXC-Ubuntu23:~#

```

Proxmox Virtual Environment 8.2.2

Contenedor 1602 (LXC-Ubuntu23-VLAN30) en el nodo vm-proxmox-01

```

root@LXC-Ubuntu23-VLAN30:~# ping 10.10.0.2
ping: connect: Network is unreachable
root@LXC-Ubuntu23-VLAN30:~# ping 10.10.0.2
ping: connect: Network is unreachable
root@LXC-Ubuntu23-VLAN30:~#

```



Salida a Internet de los contenedores

Supongamos que ahora queremos tener acceso a Internet en algún contenedor. Para ello tendremos que hacer una nueva Zona SDN Simple.

¡ATENCIÓN! esto solo funciona con la configuración de red del nodo Proxmox con Linux Bridge. No funciona para la versión de Proxmox actual (8.2.2) con OVS Bridge.

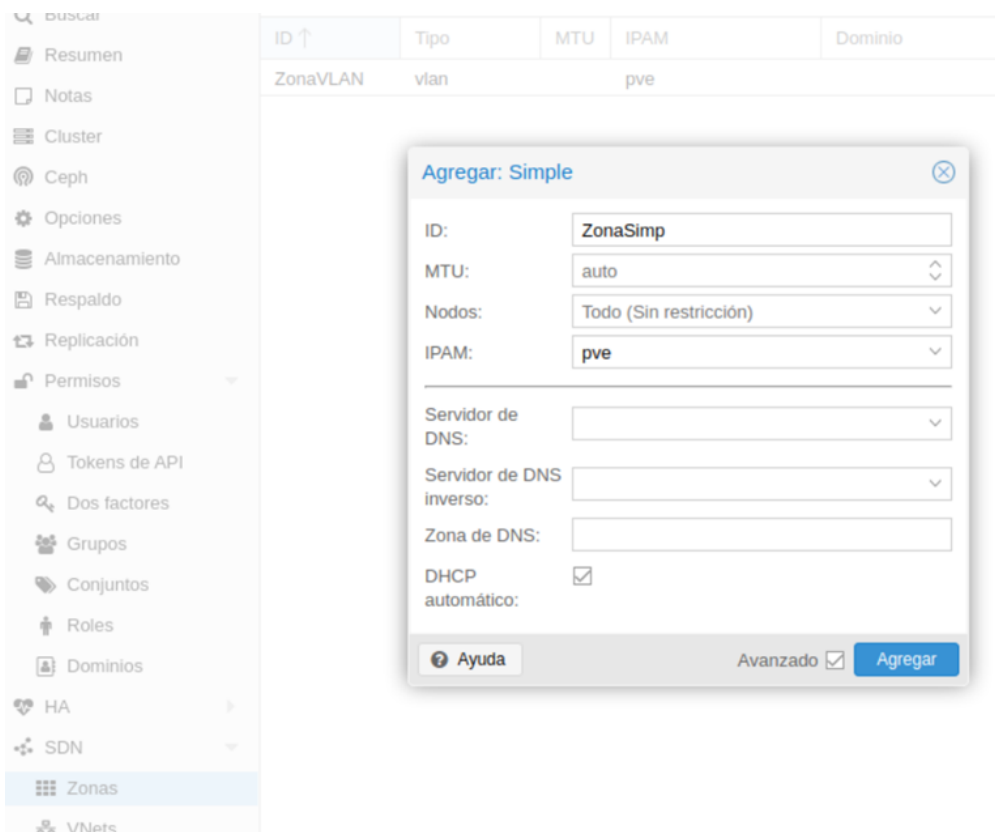


Imagen de elaboración propia: *Creación de una Zona Simple para proporcionar acceso hacia fuera de Proxmox a los contenedores y MV* ([CC BY-NC-SA](#))



The screenshot shows the Proxmox VE interface. On the left is a sidebar with a tree view of resources under 'Centro de datos' and 'vm-proxmox-01'. A central menu contains options like 'Buscar', 'Resumen', 'Notas', 'Cluster', 'Ceph', 'Opciones', 'Almacenamiento', 'Respaldo', 'Replicación', 'Permisos', 'Usuarios', 'Tokens de API', 'Dos factores', 'Grupos', and 'Conjuntos'. The main area displays a table of VNETs:

ID ↑	Alias	Zona	Etiqueta	Consci...	Estado
SwVLAN10	Switch...	ZonaV...	10		
SwVLAN30	Switch...	ZonaV...	30		

A 'Crear: VNet' dialog box is open, showing the following configuration:

- Nombre: SwSALIDA
- Alias: Switch para tener acceso a Internet
- Zona: ZonaSimp
- Etiqueta: (empty)
- Consciente de VLAN:

Buttons for 'Ayuda' and 'Crear' are visible at the bottom of the dialog.

Imagen de elaboración propia: Creación de la VNet "SwSALIDA" para proporcionar salida hacia Internet (CC BY-NC-SA)

The screenshot shows the Proxmox VE interface. The left sidebar is the same as in the previous image. The main area displays the VNETs table with an additional entry:

ID ↑	Alias	Zona	Etiqueta	Consci...	Estado
SwSALIDA	Switch...	ZonaSi...			new
SwVLAN10	Switch...	ZonaV...	10		
SwVLAN30	Switch...	ZonaV...	30		

A 'Crear: Subred' dialog box is open, showing the 'Rangos de DHCP' tab with the following configuration:

- Subred: 10.0.0.0/24
- Puerta de enlace: 10.0.0.1
- SNAT:
- Prefijo de zona de DNS: (empty)

A 'Crear' button is visible at the bottom right of the dialog.

Imagen de elaboración propia: Configuración de la Subnet para proporcionar IP por DHCP y hacer SNAT (CC BY-NC-SA)



ID ↑	Alias	Zona	Etiqueta	Consci...	Estado
SwSALIDA	Switch...	ZonaSi...			new
SwVLAN10	Switch...	ZonaV...	10		
SwVLAN30	Switch...	ZonaV...	30		

Editar: Subred

General **Rangos de DHCP**

Dirección de inicio	Dirección final	
10.0.0.2	10.0.0.100	

Agregar

Aceptar

Imagen de elaboración propia. *Configuración del DHCP* ([CC BY-NC-SA](#))

Actualizamos los cambios en la SDN.

Añadimos una nueva interfaz de red al contenedor que queremos que tenga acceso a Internet mediante Source NAT:

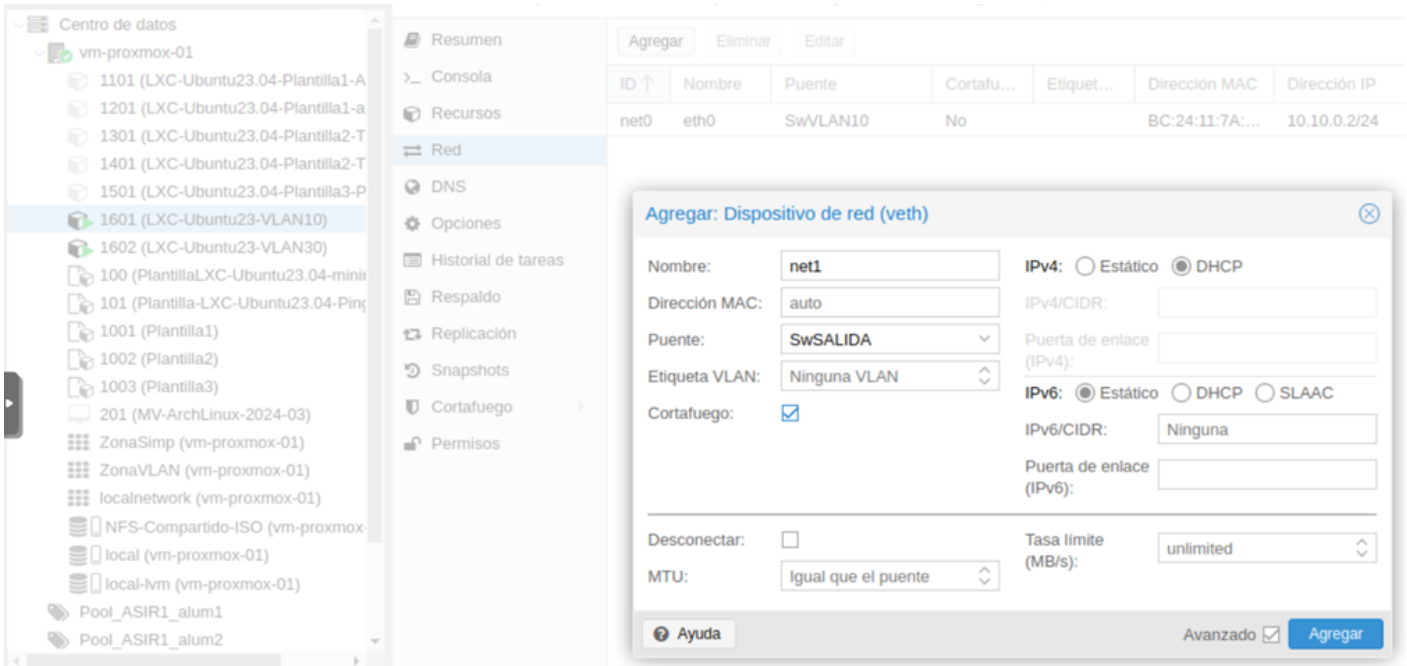


Imagen de elaboración propia: Conexión una nueva interfaz de red al "SwSALIDA" (CC BY-NC-SA)

Reiniciamos el contenedor para una correcta instalación de la tabla de enrutamiento y comprobamos las conexiones haciendo ping tanto a la zona VLAN y a google.es:

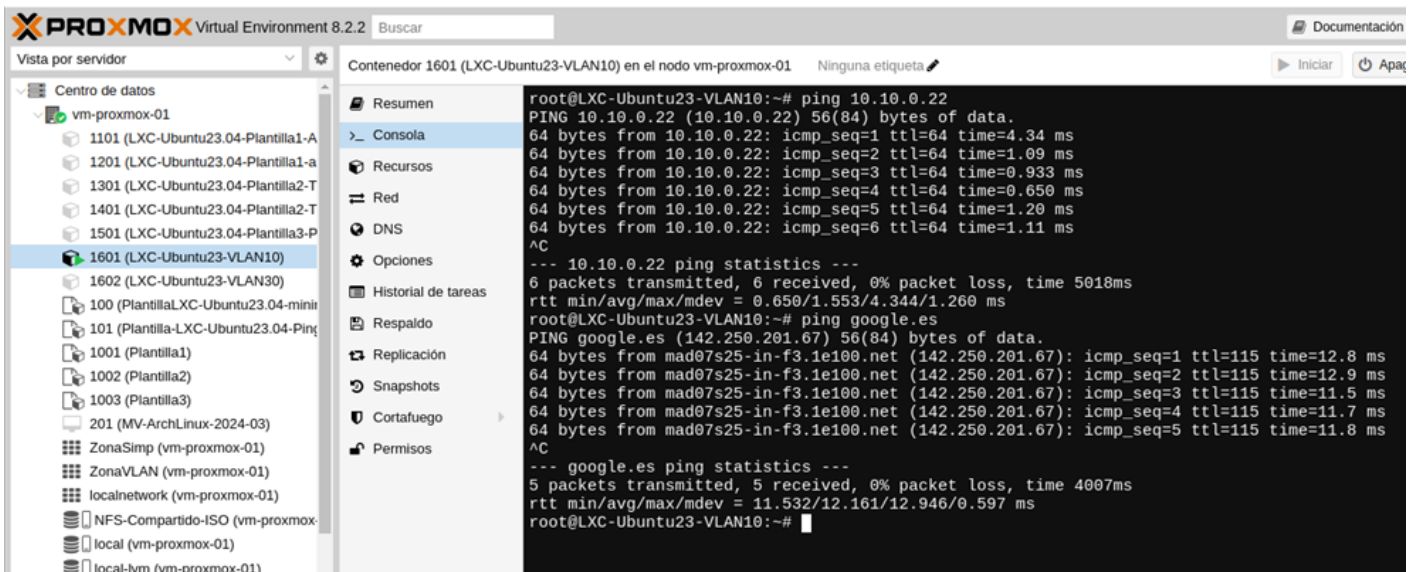


Imagen de elaboración propia: Ping al otro contenedor en la misma VLAN10 y ping a google.es (CC BY-NC-SA)



4.- SDN (Software Defined Network)

4.4.- SDN QinQ.

¿Qué es una VLAN Queue in Queue?

Queue in Queue

QinQ (conocido como apilamiento VLAN) está estandarizado por el IEEE 802.1ad. Encapsula la etiqueta VLAN con dos capas: una etiqueta interior (de una red privada) y una etiqueta exterior (de la red pública) que en nuestro caso hemos llamado datos

un paquete etiquetado 802.11Q se encapsula en otra etiqueta 802.1Q.

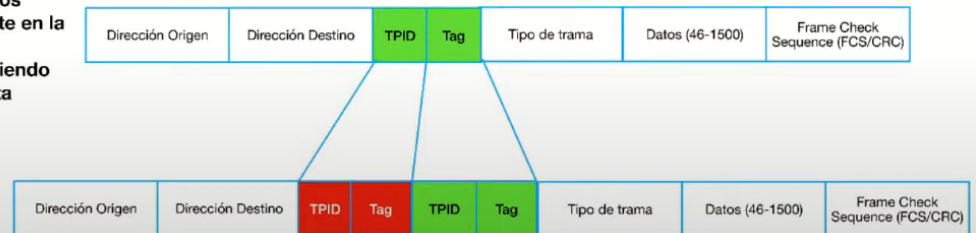
los paquetes se reenvían en función de la etiqueta VLAN exterior en la red pública, se interpreta que la etiqueta forma parte de los datos, por lo que esta también se transmite en la red pública.

Al llegar a destino se “desgrana” transmitiendo el paquete en la red destino con la etiqueta original (voz)

Trama Ethernet con VLAN (voz)



Trama Ethernet con VLAN (datos)



Eduardo Taboada (Tecnocratica.net) · VLAN Queue in Queue (Todos los derechos reservados)

La tunelización Q-in-Q en VLAN permiten crear una conexión Ethernet de capa 2 entre dos extremos y dentro del tunel podemos tener otras 4096 VLAN, es decir, lo que estamos haciendo es meter una VLAN dentro de otra VLAN.



Trama Ethernet con VLAN

Queue in Queue

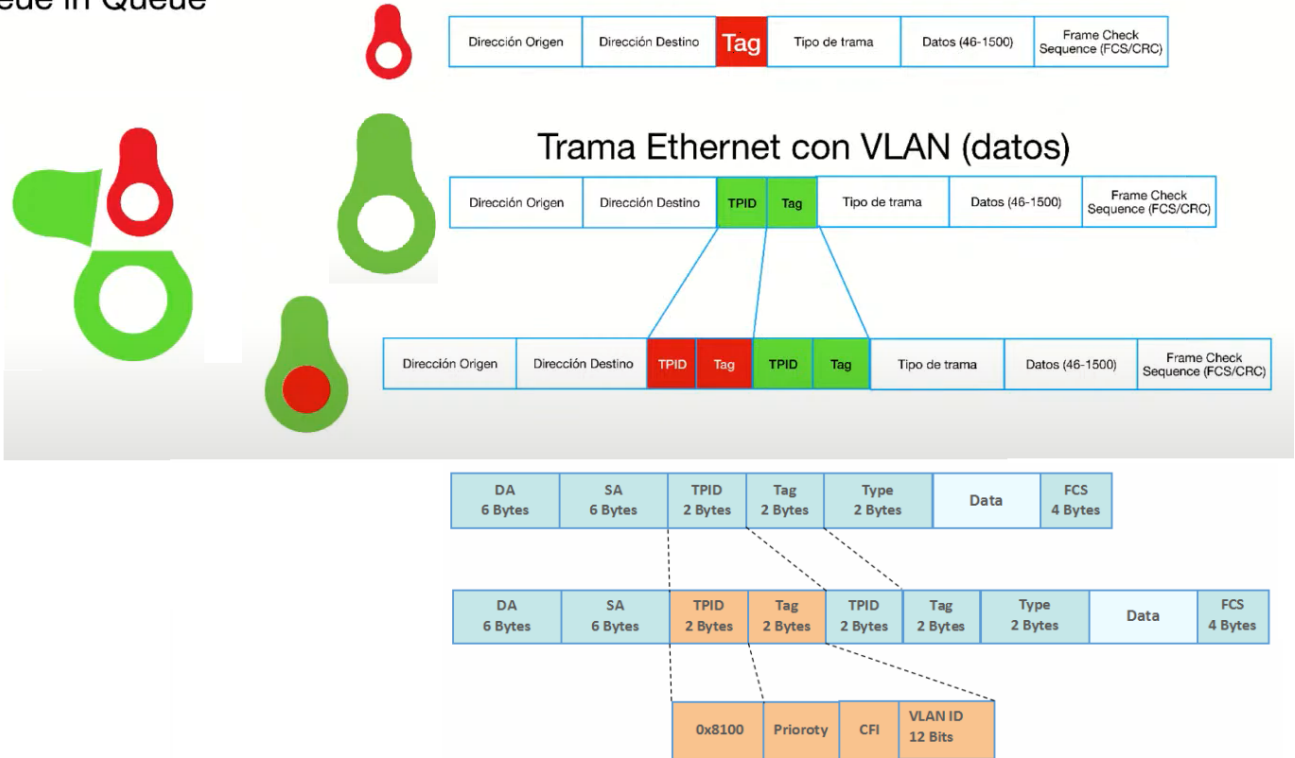


Imagen de elaboración propia: *Tunel QinQ de capa 2 y dentro una trama de VLAN de capa 2* (CC BY-NC-SA)

Q-in-Q está estandarizado por el IEEE 802.1ad. Encapsula la etiqueta VLAN con dos capas: una etiqueta interior (de una red privada) y una etiqueta exterior (de la red pública). El etiquetado VLAN tradicional que utiliza el IEEE 802.1Q es incapaz de identificar y aislar los datos de los usuarios en las tramas crecientes de Ethernet. La tecnología QinQ se utiliza para ampliar la cantidad VLANs hasta **4096x4096**, de este modo se podrán ahorrar ID de VLAN.

Los paquetes QinQ tienen un formato fijo. Normalmente, un paquete etiquetado 802.11Q se encapsula en otra etiqueta 802.1Q, de la que deriva el nombre «Q-in-Q». Durante la transmisión, los paquetes se reenvían en función de la etiqueta VLAN exterior en la red pública, se interpreta que la etiqueta forma parte de los datos, por lo que esta también se transmite en la red pública. Como contienen esta forma de doble etiqueta, los paquetes QinQ tienen 4 bytes más que los paquetes comunes con etiqueta VLAN 802.1Q.

Reduce la MTU en los vínculos de acceso en al menos 4 bytes para que las tramas no excedan la MTU del enlace de troncalización cuando se agreguen las etiquetas VLAN.

QinQ

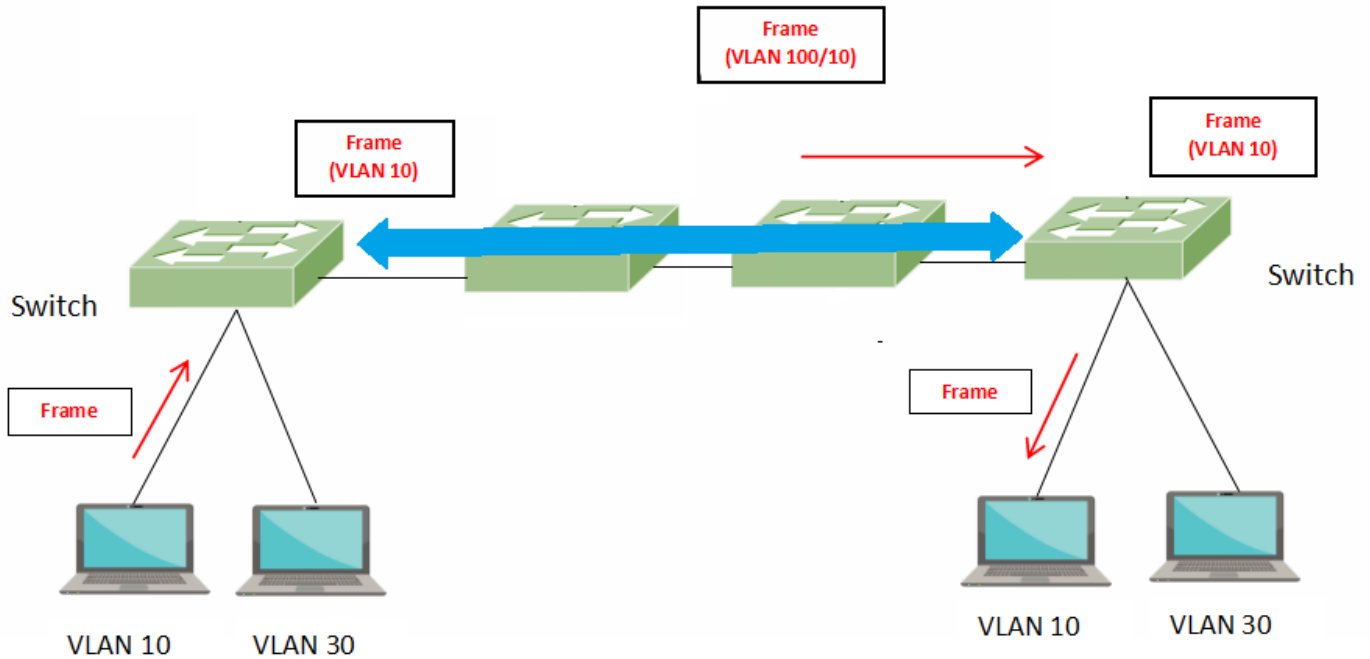


Imagen de elaboración propia. Esquema de tunelización QinQ de VLAN (CC BY-NC-SA)

Implementación de una Zona de SDN del tipo QinQ

Un caso de uso típico para esta configuración es un proveedor de hosting que proporciona una red aislada a los clientes para la comunicación de VM pero aísla las VM de otros clientes. Con QinQ cada cliente podría tener su propia VLAN (máximo de clientes 4096) pero para cada cliente se le podría proporcionar 4096 VLAN propias dentro de su tunel y aislar a su vez su tráfico de red.

Para el primer nodo Proxmox, crea una zona QinQ llamada "Qinq100" con el servicio VLAN 100 (que será la VLAN que vean todos los switch):



The screenshot shows the Proxmox VE 8.2.2 interface. On the left, the 'Centro de datos' tree is expanded to 'vm-proxmox-c01'. The 'Zonas' menu item is highlighted in the central navigation pane. A modal dialog titled 'Agregar: QinQ' is open, displaying the following configuration fields:

ID	Tipo	MTU	IPAM	Dominio
Qinq100	vmbro	100	802.1ad	auto
			Todo (Sin restricción)	
			pve	

Additional fields in the dialog include: Servidor de DNS, Servidor de DNS inverso, and Zona de DNS. The 'Avanzado' checkbox is checked, and the 'Agregar' button is visible at the bottom right of the dialog.

Imagen de elaboración propia: Creación de la Zona SDN tipo QinQ (CC BY-NC-SA)

Crea una VNet denominada "SQ10010" con VLAN-ID 10 en la zona "QinqZona100" creada anteriormente.



Centro de datos

VNets

Crear Eliminar Editar

ID ↑	Alias	Zona	Etiqueta	Consci...	Estado
------	-------	------	----------	-----------	--------

Crear: VNet

Nombre:

Alias:

Zona:

Etiqueta:

Consciente de VLAN:

[Ayuda](#)

Imagen de elaboración propia: Creación de la VNet SQ10010 con etiqueta VLAN 10 perteneciente a la Zona Qinq 100 ([CC BY-NC-SA](#))

Crea una VNet denominada "SQ10030" con VLAN-ID 30 en la zona "QinqZona100" creada anteriormente.



Centro de datos

VNets

Crear Eliminar Editar

ID ↑	Alias	Zona	Etiqueta	Consci...	Estado
SQ10010	Switch...	Qinq100	10		new

Crear: VNet

Nombre:

Alias:

Zona:

Etiqueta:

Consciente de VLAN:

Ayuda

Imagen de elaboración propia: Creación VNet SQ10030 con etiqueta VLAN 30 encapsulado en Qinq 100 ([CC BY-NC-SA](#))

Crema una zona Qinq llamada "Qinq200" con el servicio VLAN 200 (que será para otro cliente):



Agregar: QinQ ⊗

ID:	<input type="text" value="Qinq200"/>
Bridge:	<input type="text" value="vibr0"/>
Servicio VLAN:	<input type="text" value="200"/>
Protocolo del Servicio VLAN:	<input type="text" value="802.1ad"/>
MTU:	<input type="text" value="auto"/>
Nodos:	<input type="text" value="Todo (Sin restricción)"/>
IPAM:	<input type="text" value="pve"/>
Servidor de DNS:	<input type="text"/>
Servidor de DNS inverso:	<input type="text"/>
Zona de DNS:	<input type="text"/>

Avanzado

Imagen de elaboración propia: *Creación de otra Zona QinQ para comprobar el aislamiento del tráfico Ethernet* ([CC BY-NC-SA](#))

Crea una VNet denominada "SQ20010" con VLAN-ID 10 en la zona "Qinq200" creada anteriormente.

Crea una VNet denominada "SQ20030" con VLAN-ID 30 en la zona "Qinq200" creada anteriormente.

Aplica la configuración en SDN y repetir en mismo proceso en el nodo 2 de Proxmox.



PROXMOX Virtual Environment 8.2.2

Vista por servidor

- Centro de datos
 - vm-proxmox-c01
 - 100 (LXC-Ubuntu23.04-01)
 - Qinq100 (vm-proxmox-c01)
 - Qinq200 (vm-proxmox-c01)
 - localnetwork (vm-proxmox-c01)
 - NFS-Compartido-ISO (vm-proxmox-c01)
 - local (vm-proxmox-c01)
 - local-lvm (vm-proxmox-c01)
 - Pool_ASIR1_alum1
 - Pool_ASIR1_alum2
 - Pool_PROF_profe1
 - Pool_PROF_profe2
 - Pool_TIC_alum3
 - Pool_TIC_alum4

Centro de datos

- Dos factores
- Grupos
- Conjuntos
- Roles
- Dominios
- HA
- SDN
- Zonas
- VNets**
- Opciones
- IPAM

VNets

Crear Eliminar Editar

ID ↑	Alias	Zona	Etiqueta	Consci...	Estado
SQ10010	Switch...	Qinq100	10		
SQ10030	Switch...	Qinq100	30		
SQ20010		Qinq200	10		
SQ20030		Qinq200	30		

Imagen de elaboración propia: *Aplicar cambios en la SDN para crear toda la configuración de red* (CC BY-NC-SA)

← → ↻ No es seguro | <https://192.168.30.119:8006/#v1:0:18:4:::53>

PROXMOX Virtual Environment 8.2.2

Vista por servidor

- Centro de datos
 - vm-proxmox-c02
 - Qinq100 (vm-proxmox-c02)
 - Qinq200 (vm-proxmox-c02)
 - localnetwork (vm-proxmox-c02)
 - NFS-Compartido-ISO (vm-proxmox-c02)
 - local (vm-proxmox-c02)
 - local-lvm (vm-proxmox-c02)
 - Pool_ASIR1_alum1
 - Pool_ASIR1_alum2
 - Pool_PROF_profe1
 - Pool_PROF_profe2
 - Pool_TIC_alum3
 - Pool_TIC_alum4

Centro de datos

- Roles
- Dominios
- HA
- SDN**
- Zonas
- VNets
- Opciones
- IPAM
- ACME
- Cortafuego

Estado

Aplicar

SDN	Nodo	Estado
localnet...	vm-proxmox-c02	ok
Qinq100	vm-proxmox-c02	available
Qinq200	vm-proxmox-c02	available

Imagen de elaboración propia: *Creación de las dos Qinq en el nodo 2 de Proxmox al igual que hemos hecho en el nodo 1* (CC BY-NC-SA)

Crea 4 contenedores en el nodo 1 de Proxmox, asignándoles IP según la siguiente tabla:



CLIENTE CPD	ID CONTENEDOR	IP CONTENEDOR	Zona QinQ	VNet VLAN
1	1001	10.0.1.110/16	QinqZona100	SQ100-10
	1002	10.0.1.130/16		SQ100-30
2	2001	10.0.1.210/16	QinqZona200	SQ200-10
	2002	10.0.1.230/16		SQ200-30

Ahora, crea 4 contenedores en el nodo 2 de Proxmox, asignándoles IP según la siguiente tabla:

CLIENTE CPD	ID CONTENEDOR	IP CONTENEDOR	Zona QinQ	VNet VLAN
1	1021	10.0.2.110/16	QinqZona100	SQ100-10
	1022	10.0.2.130/16		SQ100-30
2	2021	10.0.2.210/16	QinqZona200	SQ200-10
	2022	10.0.2.230/16		SQ200-30

De tal manera que el CT1001 solo puede hacer ping al CT1021 y viceversa, y no podrá hacer ping al resto de contenedores porque se encuentra el tráfico de tramas Ethernet aislado por su QinQ:

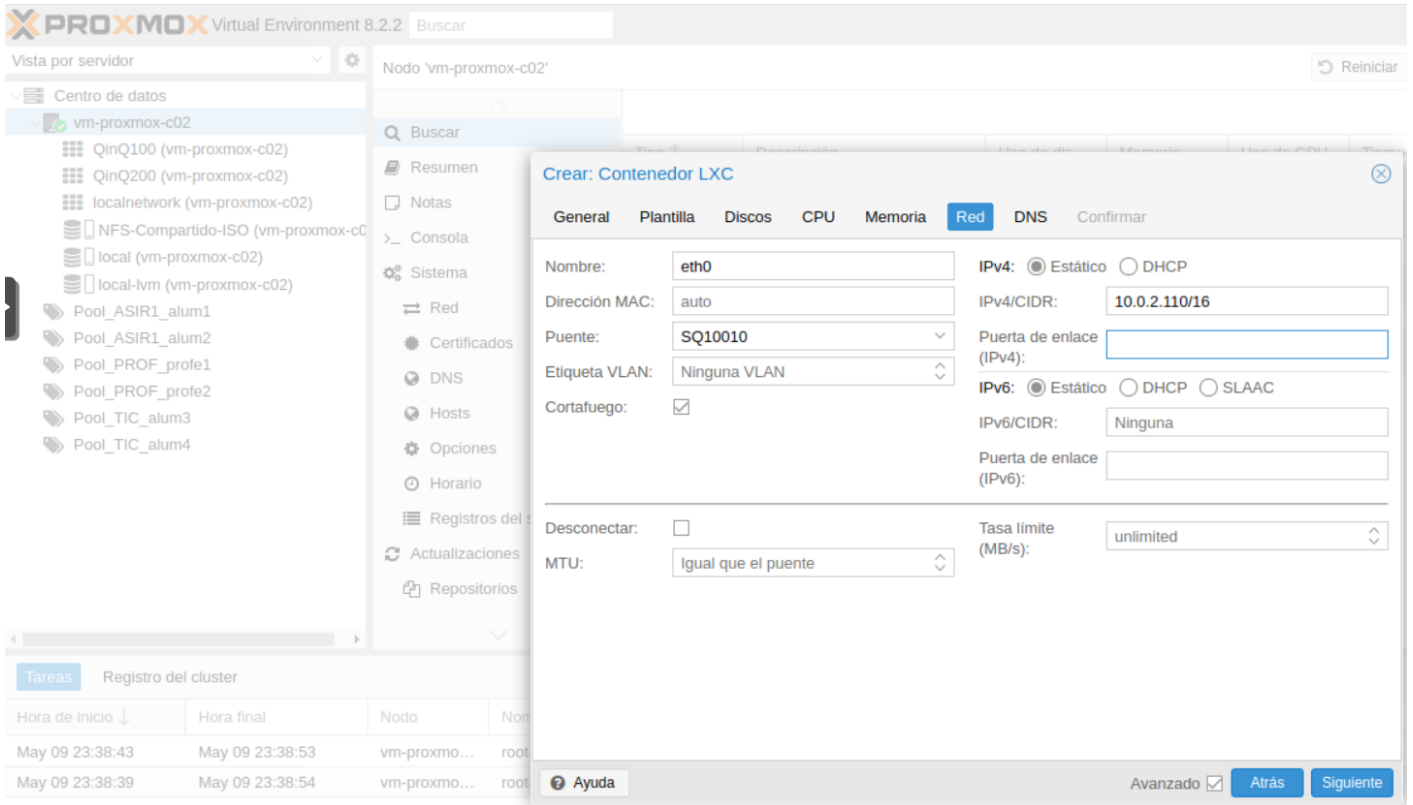


Imagen de elaboración propia: Creación del contenedor 1021 (CC BY-NC-SA)

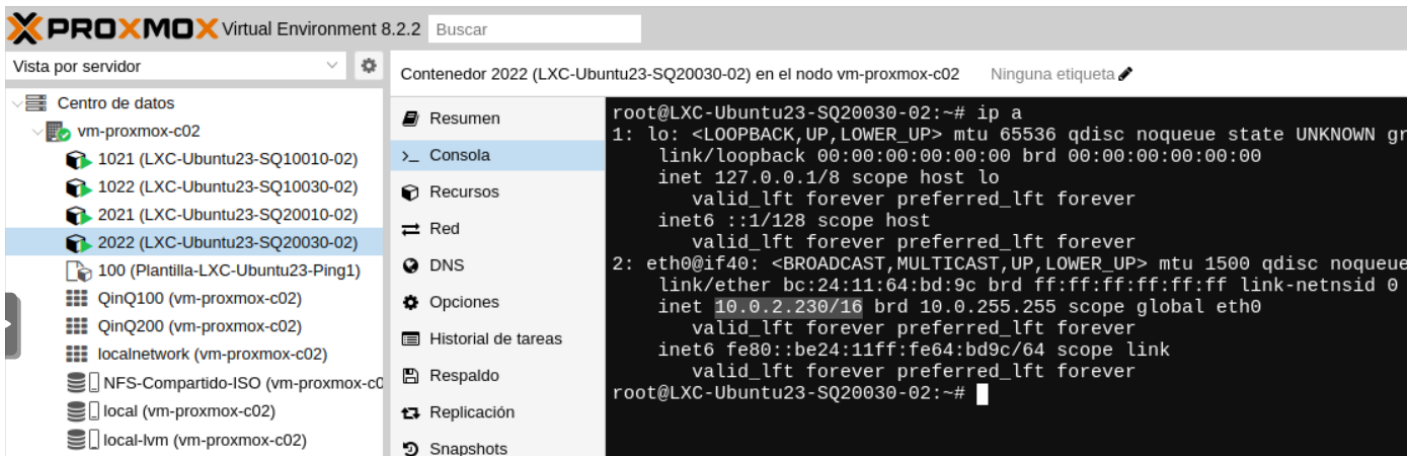


Imagen de elaboración propia: Configuración de red del contenedor 2022 (CC BY-NC-SA)

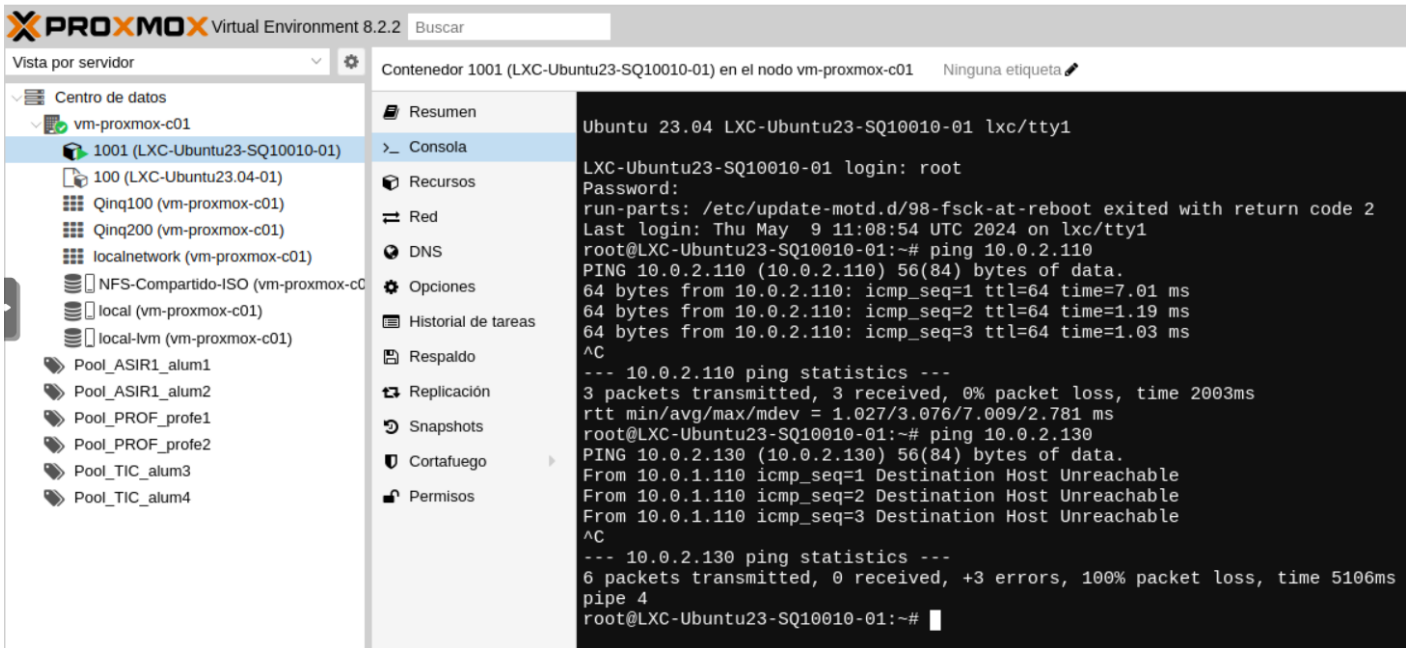


Imagen de elaboración propia: Ping desde el CT 1010 del nodo 1 al CT 2010 del nodo 2 y rechazados todos los demás (CC BY-NC-SA)

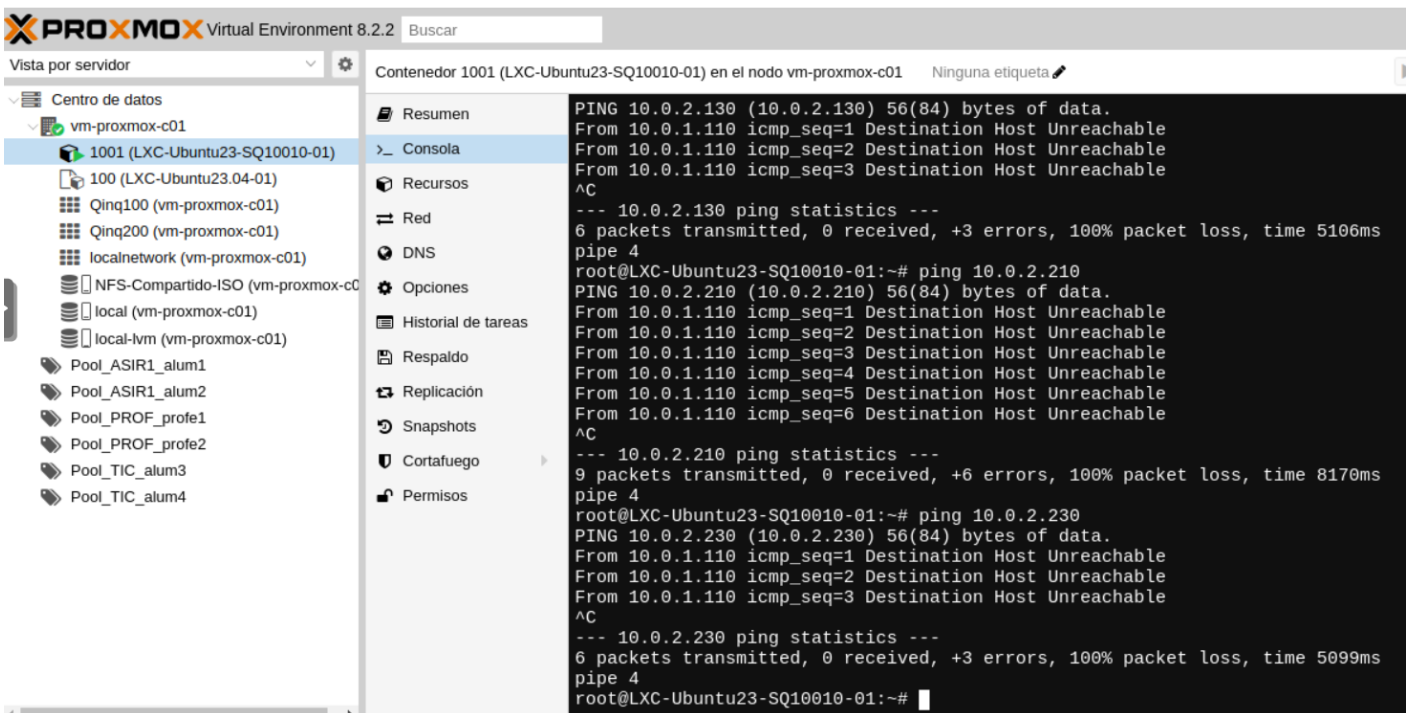


Imagen de elaboración propia: Rechazados todos los ping al resto de CT tal y cómo se esperaba (CC BY-NC-SA)



4.- SDN (Software Defined Network)

4.5.- SDN VxLAN.

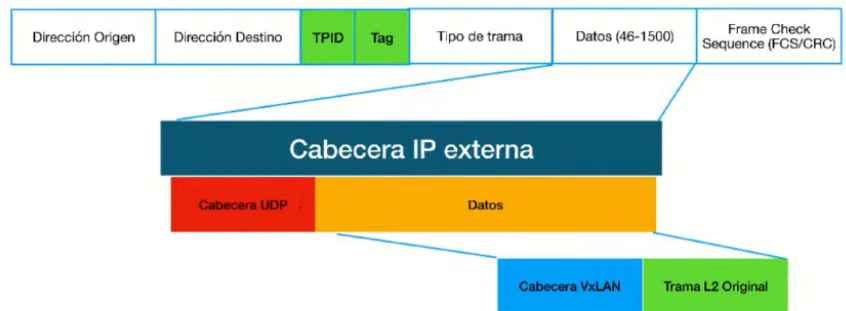
¿Qué es una VxLAN?

VxLAN

La VxLAN (LAN extensible virtual), permite extender las redes virtuales más allá de los límites de una sola red física. Funciona encapsulando los paquetes de red en otro paquete. Permite extender una red de Nivel o capa 2 en una red más extensa de Capa3. Permite encapsular el protocolo de dirección MAC (layer2) en datagramas de usuario (MAC-in-UDP).

Con la encapsulación VxLAN MAC-in-UDP, los paquetes originales se añadirán en una cabecera VxLAN y luego se colocarán en un paquete UDP-IP

Trama Ethernet



Trama Ethernet con VLAN (datos)



Eduardo Taboada (Tecnocratica.net) VxLAN (Todos los derechos reservados)

VxLAN (Red de área local virtual **extensible**) es una tecnología de superposición para la virtualización de redes, que establece un **túnel lógico en la red IP para extender la red de capa 2 sobre una red subyacente de capa 3 existente**. VxLAN utiliza el Punto de Túnel VxLAN (VTEP), que puede ser un host final o switches de red, o enrutadores, para encapsular y desencapsular el tráfico de capa 2.

VxLAN utiliza UDP (puerto por defecto 4789) de capa 4 por lo que no se preocupa de confiabilidad de la transmisión, por ello, pueden perderse paquetes con tramas VLAN dentro de este, pero serán los extremos de la VxLAN quien deban de solicitar la retransmisión del tráfico perdido.

VxLAN se estandariza como un protocolo de encapsulación de superposición. Aumenta la escalabilidad hasta **16 millones redes lógicas** y permite la adyacencia de capa 2 a través de redes IP.



VxLAN es un protocolo de túnel IP estándar para ampliar las VLAN en una red. Conecta las VLAN de un extremo a otro de la red sin tunelización, es decir, los paquetes IP entre routers no van cifrados y por lo tanto, están expuesto a un sniffer de red. **VxLAN no debe ser utilizado en redes públicas.**

Para solucionar el problema de seguridad, Proxmox ha añadido la posibilidad de utilizar (que no viene por defecto) **VxLAN IPSEC Encryption, con un cifrado AES 256-sha1**. Para agregar cifrado IPSEC en una VxLAN, deberá reducir la MTU en 60 bytes adicionales para IPv4 u 80 bytes para IPv6 para manejar el cifrado. Entonces, con un MTU de 1500 reales predeterminados, debe usar un MTU de 1370 (1370+80(IPSEC)+50(VXLAN)==1500).

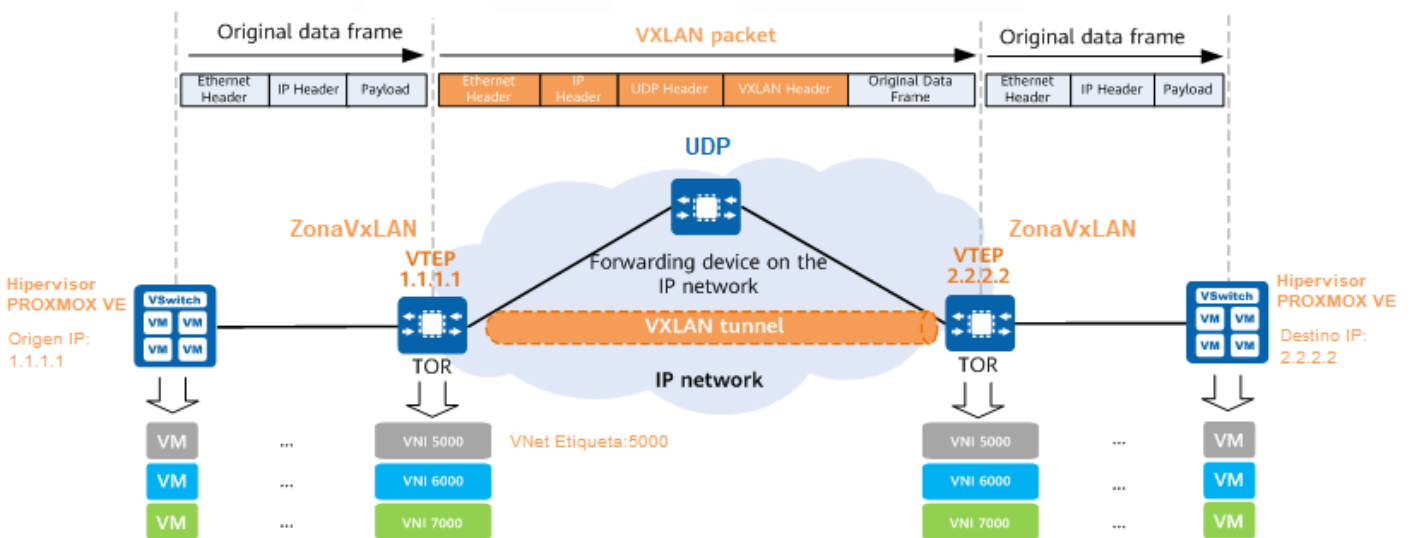


Imagen de elaboración propia: Esquema VxLAN en Proxmox VE 8.2 (CC BY-NC-SA)

Implementar una Zona SDN del tipo VxLAN entre dos o más nodos Proxmox

Realizaremos una VxLAN entre dos nodos Proxmox, pero se pueden unir más, con las direcciones IP de nodo 192.168.30.221, 192.168.30.119.

Crearemos una zona VxLAN llamada ZonVxLAN, agrega todas las IP de los nodos a la lista de direcciones de pares. Utilizaremos una MTU predeterminada de 1450 en los contenedores o MV.



← → ↻ No es seguro | https://192.168.30.221:8006/#v1:0:18:4::=contentVztmpl::=consolejs:=sdnzone

PROXMOX Virtual Environment 8.2.2

Vista por servidor

- Centro de datos
 - vm-proxmox-c01
 - 1001 (LXC-Ubuntu23-SQ10010-01)
 - 100 (LXC-Ubuntu23.04-01)
 - Qinq100 (vm-proxmox-c01)
 - Qinq200 (vm-proxmox-c01)
 - localnetwork (vm-proxmox-c01)
 - NFS-Compartido-ISO (vm-proxmox-c01)
 - local (vm-proxmox-c01)
 - local-lvm (vm-proxmox-c01)
 - Pool_ASIR1_alum1
 - Pool_ASIR1_alum2
 - Pool_PROF_profe1
 - Pool_PROF_profe2
 - Pool_TIC_alum3
 - Pool_TIC_alum4

Centro de datos

- Replicación
- Permisos
- Usuarios
- Tokens de API
- Dos factores
- Grupos
- Conjuntos
- Roles
- Dominios
- HA
- SDN
- Zonas**
- VNets
- Opciones
- IPAM
- ACME

ID ↑	Tipo	MTU	IPAM	Dominio
Qinq100	qinq		pve	
Qinq200	qinq		pve	
ZonVxLAN				

Editar: VXLAN
ID: ZonVxLAN
Lista de direcciones de pares:
MTU:
Nodos:
IPAM:
Servidor de DNS:
Servidor de DNS inverso:
Zona de DNS:
 Avanzado

Imagen de elaboración propia: Creación de la Zona SDN de tipo VxLAN entre dos nodos Proxmox ([CC BY-NC-SA](#))

Crea una VNet denominada "SwVxLAN1" utilizando la zona VXLAN "ZonVxLAN" creada anteriormente, con etiqueta 300:



ID	Alias	Zona	Etiqueta	Consci...	Estado
SQ10010	Switch...	Qinq100	10		
SQ10030	Switch...	Qinq100	30		
SQ20010		Qinq200	10		
SQ20030		Qinq200	30		

Imagen de elaboración propia: Creación de VNet "SwVxLAN1" (CC BY-NC-SA)

Aplica la configuración en la SDN para crear redes virtuales y crea un contenedor uniendo su interfaz de red al SwVxLAN1 con una MTU de 1450 y una IP estática 10.0.4.100/24

<https://192.168.30.119:8006/#v1:0:=lxc%2F4002:4:::11:=sdnvnet>

Nombre:	eth0	IPv4:	<input checked="" type="radio"/> Estático <input type="radio"/> DHCP
Dirección MAC:	BC:24:11:7F:9C:D3	IPv4/CIDR:	10.0.4.101/24
Puente:	SwVxLAN1	Puerta de enlace (IPv4):	
Etiqueta VLAN:	Ninguna VLAN	IPv6:	<input checked="" type="radio"/> Estático <input type="radio"/> DHCP <input type="radio"/> SLAAC
Cortafuego:	<input checked="" type="checkbox"/>	IPv6/CIDR:	Ninguna
Desconectar:	<input type="checkbox"/>	Puerta de enlace (IPv6):	
MTU:	1450	Tasa límite (MB/s):	unlimited



Imagen de elaboración propia: Configuración de red de un contenedor en la VxLAN en el nodo c01 de Proxmox (CC BY-NC-SA)

Ahora, haremos lo mismos pasos en el nodo c02 de Proxmox:

Centro de datos

- vm-proxmox-c02
 - 1021 (LXC-Ubuntu23-SQ10010-02)
 - 1022 (LXC-Ubuntu23-SQ10030-02)
 - 2021 (LXC-Ubuntu23-SQ20010-02)
 - 2022 (LXC-Ubuntu23-SQ20030-02)
 - 100 (Plantilla-LXC-Ubuntu23-Ping1)
 - QinQ100 (vm-proxmox-c02)
 - QinQ200 (vm-proxmox-c02)
 - ZonVxLAN (vm-proxmox-c02)
 - localnetwork (vm-proxmox-c02)
 - NFS-Compartido-ISO (vm-proxmox-c02)

Centro de datos

- Conjuntos
- Roles
- Dominios
- HA
- SDN**
- Zonas
- VNets
- Opciones

Estado

SDN	Nodo	Estado
localnetwork	vm-proxmox-c02	ok
QinQ100	vm-proxmox-c02	available
QinQ200	vm-proxmox-c02	available
ZonVxLAN	vm-proxmox-c02	available

Imagen de elaboración propia: Creación de Zona SDN del tipo VxLAN en el nodo c02 de Proxmox (CC BY-NC-SA)

https://192.168.30.119:8006/#v1:0:=lxc%2F4002:4:11:=sdnvnnet

Environment 8.2.2

Contenedor 4002 (LXC-Ubuntu23-SQ10010-02)

Editar: Dispositivo de red (veth)

Nombre:

Dirección MAC:

Puente:

Etiqueta VLAN:

Cortafuego:

Desconectar:

MTU:

IPv4: Estático DHCP

IPv4/CIDR:

Puerta de enlace (IPv4):

IPv6: Estático DHCP SLAAC

IPv6/CIDR:

Puerta de enlace (IPv6):

Tasa límite (MB/s):

Avanzado



Imagen de elaboración propia: Configuración de red del CT 4002 en el nodo c02 de Proxmox (CC BY-NC-SA)

Comprobaremos la interconexión entre contenedores de la VxLAN:

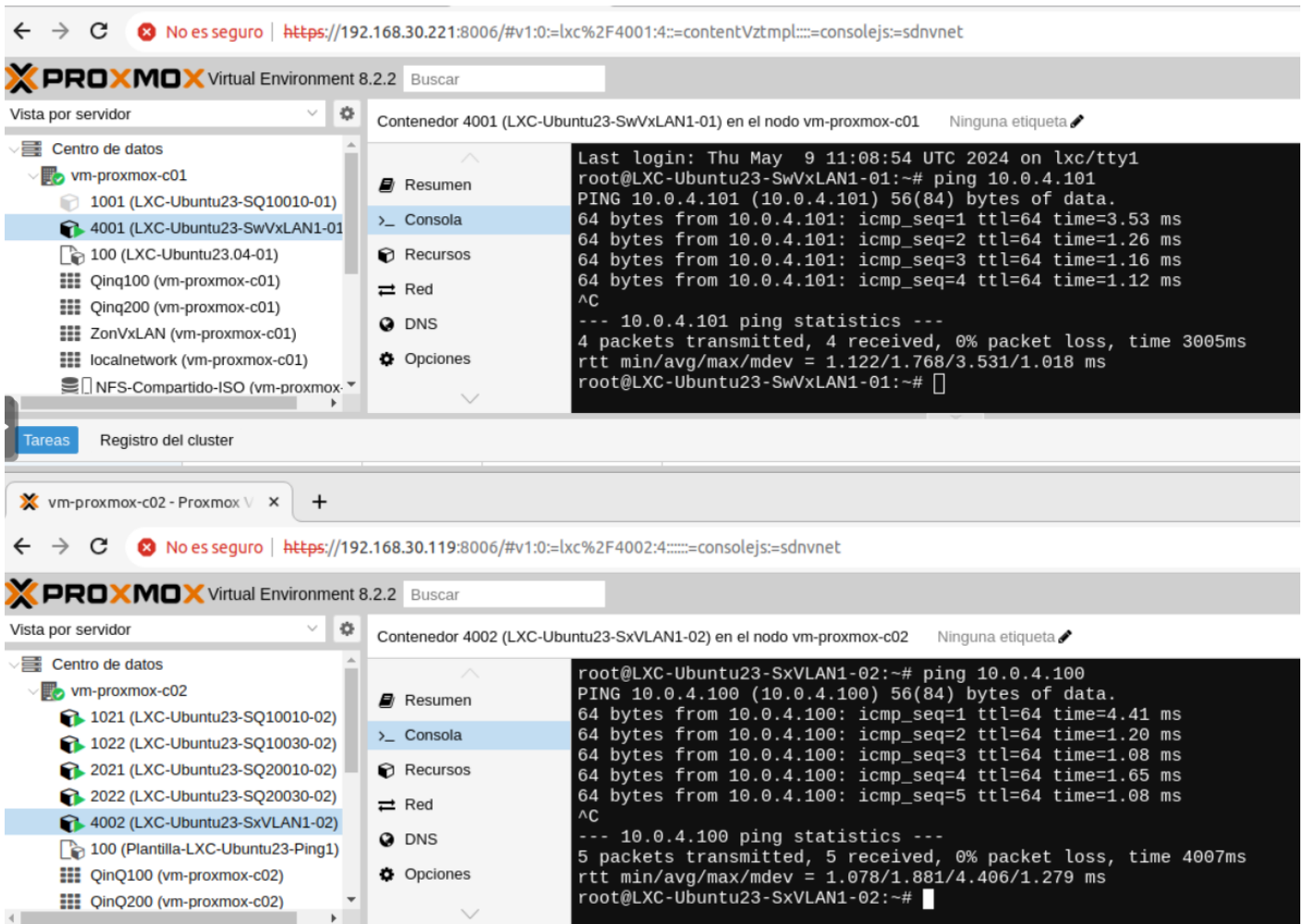


Imagen de elaboración propia: Ping entre CT4001 y CT4002 que se encuentran en dos nodos Proxmox distintos (CC BY-NC-SA)

Ahora solo nos quedaría cifrar el tunel. Proxmox ha añadido la posibilidad de utilizar (que no viene por defecto) VxLAN IPSEC Encryption, con un cifrado AES 256-sha1. Para agregar cifrado IPSEC en una VxLAN, deberá reducir la MTU en 60 bytes adicionales para IPv4 u 80 bytes para IPv6 para manejar el cifrado. Entonces, con un MTU de 1500 reales predeterminados, debe usar un MTU de 1370 (1370+80(IPSEC)+50(VXLAN)==1500).



PROXMOX Virtual Environment 8.2.2

Vista por servidor

Contenedor 4001 (LXC-Ubuntu23-SwVxLAN1-01) en el nodo vm-proxmox-c01 Ninguna etiqueta

Resumen

Consola

Recursos

Red

DNS

Opciones

Historial de tareas

Respaldo

Replicación

Snapshots

Cortafuego

Permisos

ID ↑	Nombre	Puente	Cortafu...	Etiquet...	Dirección MAC	Dirección IP
net0	eth0	SwVxL...	Sí		BC:24:11:05:...	10.0.4.100/24

Editar: Dispositivo de red (veth)

Nombre: IPv4: Estático DHCP

Dirección MAC: IPv4/CIDR:

Puente: Puerta de enlace (IPv4):

Etiqueta VLAN: IPv6: Estático DHCP SLAAC

Cortafuego: IPv6/CIDR:

Puerta de enlace (IPv6):

Desconectar: Tasa límite (MB/s):

MTU:

Avanzado

Imagen de elaboración propia: Cambio el MTU a 1370 en cada contenedor o MV pertenecientes a la VxLAN (CC BY-NC-SA)

En primer lugar debemos instalar el paquete "strongswan" en los nodos Proxmox de los extremos de los pares de la VxLAN:

```
apt install strongswan
```

Debemos modificar el fichero de configuración **/etc/ipsec.conf**. Cifraremos el tráfico UDP por el puerto 4789 que es el utilizado por VxLAN.

```

conn %default
    ike=aes256-sha1-modp1024! # the fastest, but reasonably secure cipher on modern HW
    esp=aes256-sha1!
    leftfirewall=yes        # this is necessary when using Proxmox VE firewall rules

conn output
    rightsubnet=%dynamic[udp/4789]
    right=%any
    type=transport

```



```
authby=psk
```

```
auto=route
```

```
conn input
```

```
leftsubnet=%dynamic[udp/4789]
```

```
type=transport
```

```
authby=psk
```

```
auto=route
```

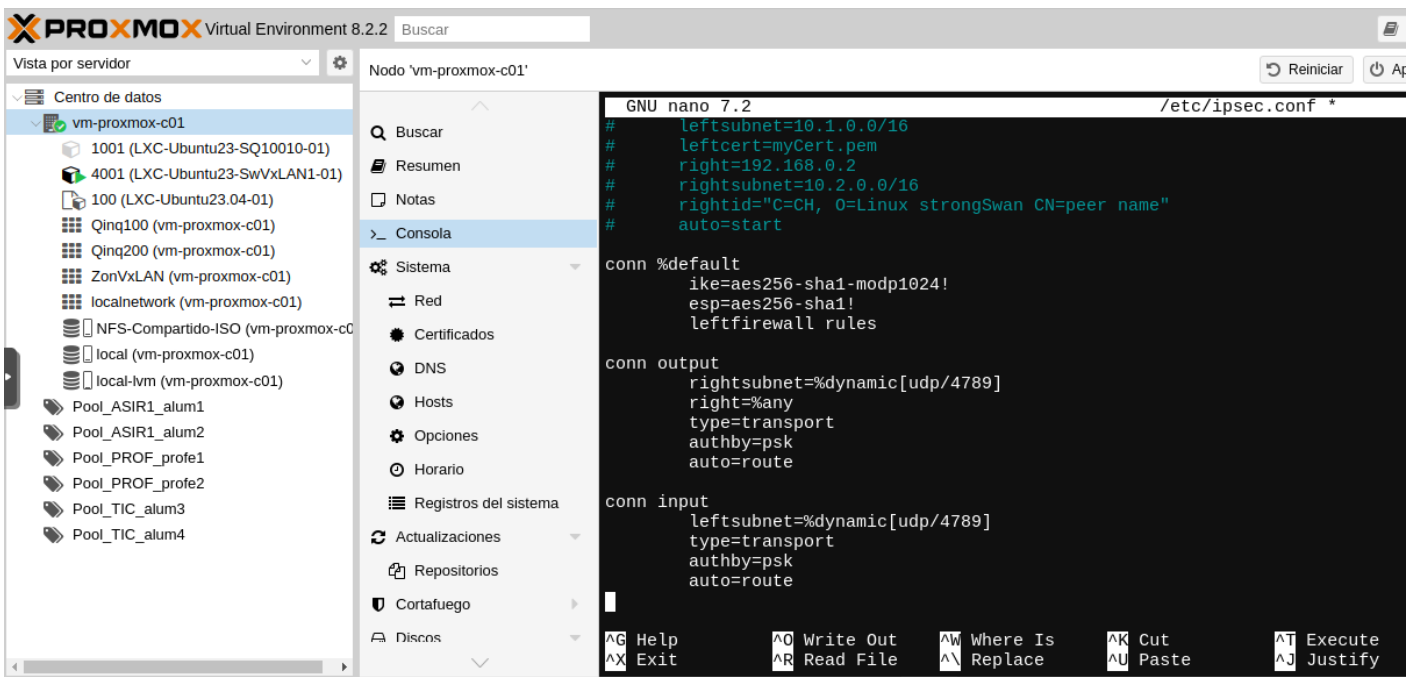


Imagen de elaboración propia: *Modificación del fichero /etc/ipsec.conf* ([CC BY-NC-SA](#))

Tenemos que generar una clave pre-compartida, para poder comenzar la negociación de seguridad en el tunel:

```
openssl rand -base64 128
```

y añadimos la clave al fichero /etc/ipsec.secrets

```
: PSK <generatedbase64key>
```

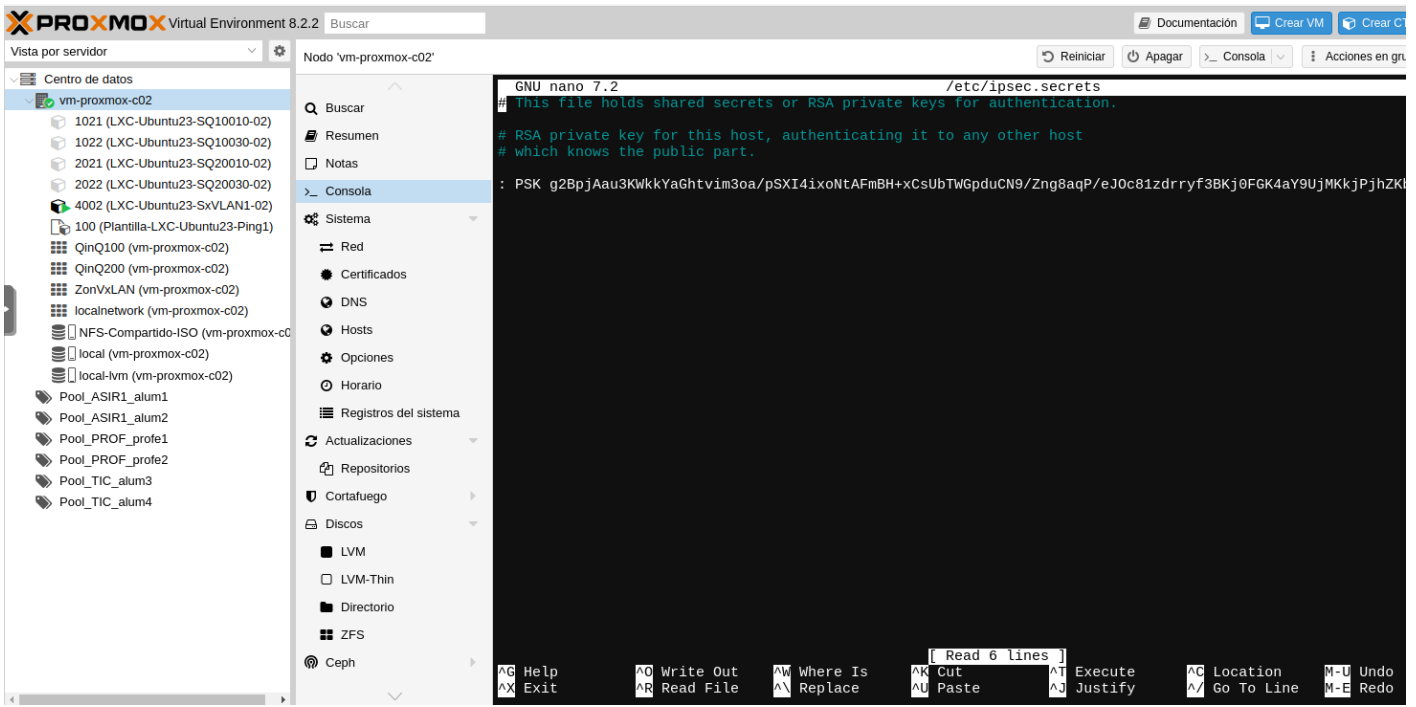


Imagen de elaboración propia: *Modificación del fichero /etc/ipsec.secrets* (CC BY-NC-SA)

Copia los dos ficheros en todos los nodo de Proxmox que formen parte de la VxLAN.



4.- SDN (Software Defined Network)

4.6.- SDN EVPN.

¿Qué es una EVPN?

En primer lugar, debemos aclarar que **VXLAN-BGP-EVPN** se refiere a una práctica de comunicación VXLAN basada en BGP EVPN. VXLAN-BGP-EVPN puede descubrir y establecer túneles automáticamente, lo que permite una migración ilimitada y sin problemas de las máquinas virtuales en el centro de datos sin que el usuario lo perciba.

BGP (Border Gateway Protocol) es el principal protocolo que soporta Internet y se utiliza para sincronizar la información de enrutamiento entre los routers.

EVPN es una extensión de BGP, que proporciona principalmente el reenvío de rutas múltiples a través del modelo de conexión múltiple (multi-homing). Su redundancia permite que un dispositivo se conecte a dos o más dispositivos ascendentes y utilice todos los enlaces para el reenvío de tráfico.

Ethernet VPN (EVPN) se basa en un modelo de VPN clásico que utiliza las extensiones de BGP MP. El concepto de una instancia VRF (enrutamiento virtual y reenvío) se hereda del mundo L3VPN/L2VPN en EVPN.

EVPN-VXLAN permite a las empresas conectar ubicaciones geográficamente dispersas mediante la creación de puentes virtuales de capa 2. EVPN-VXLAN proporciona la escala requerida por los proveedores de servicios de nube y, a menudo, también es la tecnología preferida para las interconexiones de centros de datos.

En el marco **VXLAN** inicial (definido en RFC 7348), no hay un plano de control, los túneles VXLAN se configuran manualmente y el descubrimiento de VTEP y el **aprendizaje de la información del host se realizan mediante inundación de tráfico en el plano de datos**. La información del host incluye direcciones IP, direcciones MAC, VNI y direcciones IP VTEP de puerta de enlace. Este marco es fácil de implementar, pero **genera una gran cantidad de tráfico en la red** y complica la expansión de la red. Para resolver los problemas anteriores, **VXLAN introduce EVPN como su plano de control**. Específicamente, después de implementar EVPN, **VXLAN usa rutas EVPN para transmitir direcciones VTEP e información del host**, moviendo el descubrimiento de VTEP y el aprendizaje de información del host desde el plano de datos al plano de control.



EVPN puede anunciar tanto información de dirección MAC de Capa 2 como información de ruta IP de Capa 3.

Cuando se utiliza EVPN para establecer dinámicamente un túnel VXLAN, dos VTEP establecen una relación de pares BGP EVPN e intercambian rutas de tipo 3 para transmitir información de direcciones IP VNI y VTEP para el establecimiento del túnel VXLAN.

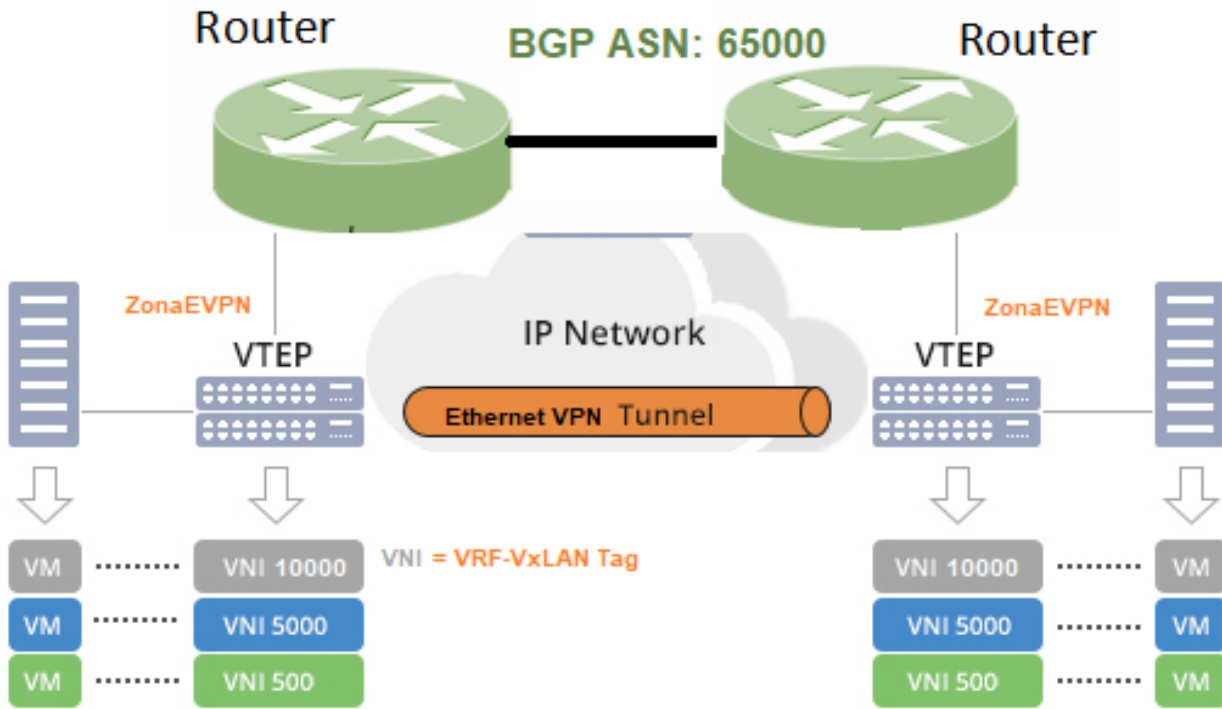


Imagen de elaboración propia: Esquema de una EVPN (CC BY-NC-SA)

Implementación de una Zona SDN del tipo EVPN entre dos nodos Proxmox

El requisito previo es tener instalado en los nodos Proxmox el paquete para enrutamiento **FRRouting**, que utiliza el protocolo BGP:

```
apt install frr-pythontools
```



El ejemplo supone un dos nodos Proxmox (vm-proxmox-c01 y vm-proxmox-c02 pero podrían ser más nodos) con direcciones IP 192.168.30.221, 192.168.30.119.

Creo un controlador EVPN utilizando un número ASN privado y las direcciones de nodo anteriores como pares.

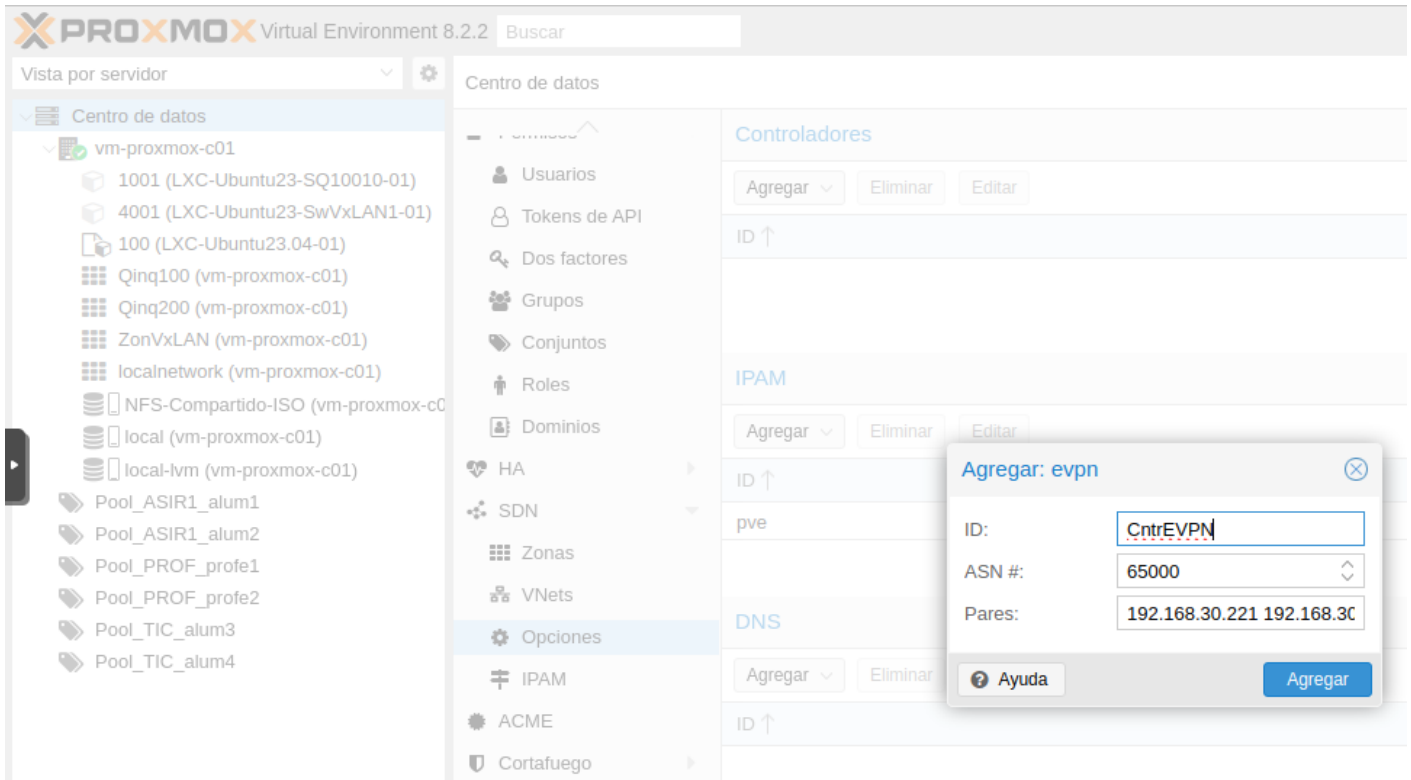


Imagen de elaboración propia: Creación de un controlador del tipo EVPN (CC BY-NC-SA)

Creo una zona EVPN llamada CntrEVPN , asigne el controlador EVPN creado previamente con etiqueta VRF-VxLAN 10000



Virtual Environment 8.2.2

Vista por servidor

Centro de datos

Centro de datos

- vm-proxmox-c01
 - 1001 (LXC-Ubuntu23-SQ10010-01)
 - 4001 (LXC-Ubuntu23-SwVxLAN1-01)
 - 100 (LXC-Ubuntu23.04-01)
 - Qinq100 (vm-proxmox-c01)
 - Qinq200 (vm-proxmox-c01)
 - ZonVxLAN (vm-proxmox-c01)
 - localnetwork (vm-proxmox-c01)
 - NFS-Compartido-ISO (vm-proxmox-c01)
 - local (vm-proxmox-c01)
 - local-lvm (vm-proxmox-c01)
 - Pool_ASIR1_alum1
 - Pool_ASIR1_alum2
 - Pool_PROF_profe1
 - Pool_PROF_profe2
 - Pool_TIC_alum3
 - Pool_TIC_alum4

Usuarios

Tokens de API

Dos factores

Grupos

Conjuntos

Roles

Dominios

HA

SDN

Zonas

VNets

Opciones

IPAM

ACME

Cortafuego

Agregar

Eliminar

ID ↑

Tipo

Qinq100

qinq

Qinq200

qinq

ZonVxLAN

vxlan

Agregar: EVPN

ID: ZonaEVPN

Controlador: CntrEVPN

VRF-VXLAN Tag: 10000

Dirección MAC de la VNet: auto

Salir de nodos: vm-proxmox-c01

Nodo primario de salida: vm-proxmox-c01

Salir del enrutador local de nodos:

Anunciar subredes:

Desactivar suspensión de ARP-nd:

Importar ruta de destino:

MTU: 1450

Nodos: Todo (Sin restricción)

IPAM: pve

Servidor de DNS:

Servidor de DNS inverso:

Zona de DNS:

Ayuda

Avanzado

Agregar

Tareas

Registro del cluster

Hora de inicio ↓	Hora final	Nodo	Nombre de usuario	Descripción
May 10 10:23:21	May 10 10:23:25	vm-proxmox...	root@pam	CT 4001 - App
May 10 10:23:15	May 10 10:23:25	vm-proxmox...	root@pam	VM/CT 4001 -
May 10 10:11:49	May 10 10:11:56	vm-proxmox...	root@pam	SRV network
May 10 10:11:46	May 10 10:11:56	vm-proxmox...	root@pam	reloadnetwork
May 10 10:08:46	May 10 10:11:36	vm-proxmox...	root@pam	Consola

Imagen de elaboración propia: Creación de una Zona SDN del tipo EVPN (CC BY-NC-SA)

Crea la primera VNet denominada sEVPN1 utilizando la zona EVPN ZonaEVPN con etiqueta 11000:

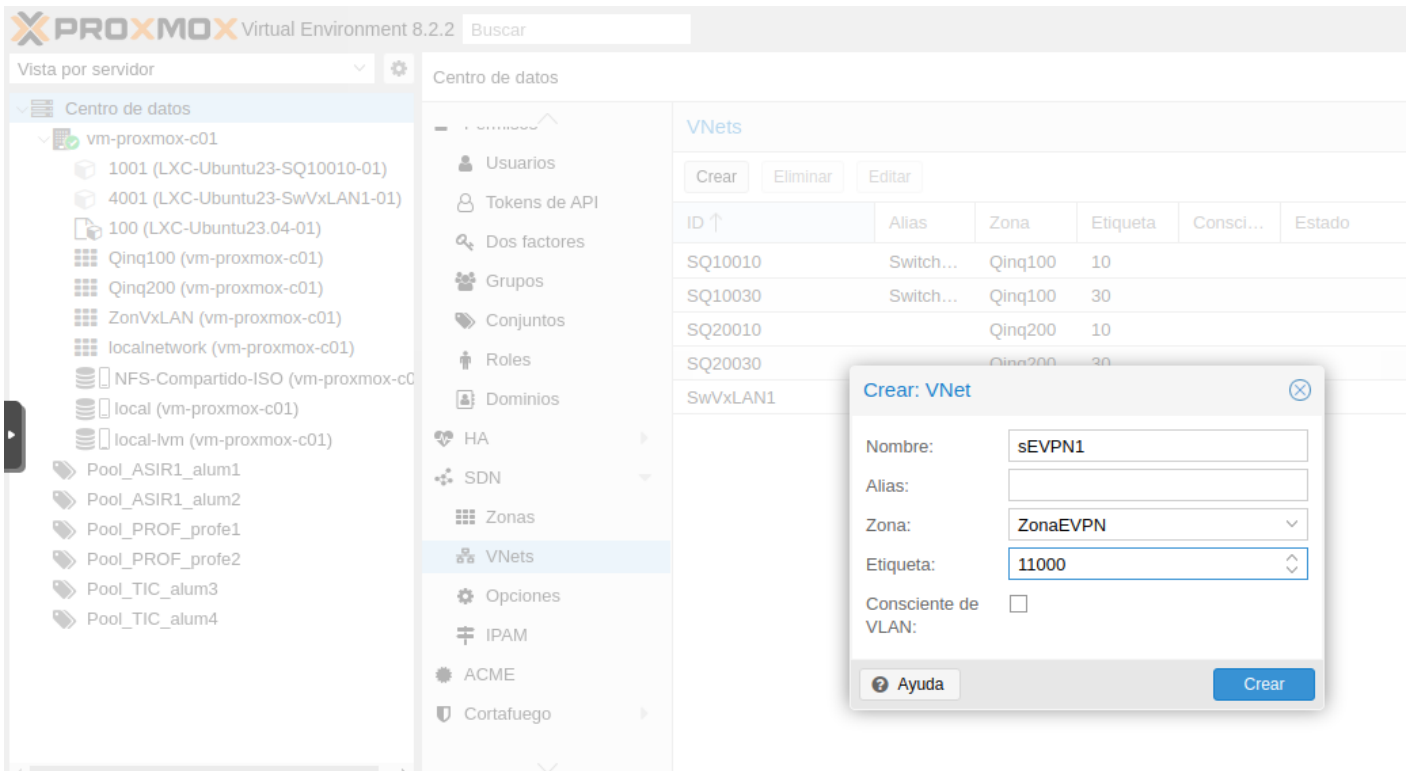


Imagen de elaboración propia: Crear la VNet llamada "sEVPN1" (CC BY-NC-SA)

Crea una subred en sEVPN1 10.0.1.0/24 y puerta de enlace 10.0.1.1 :

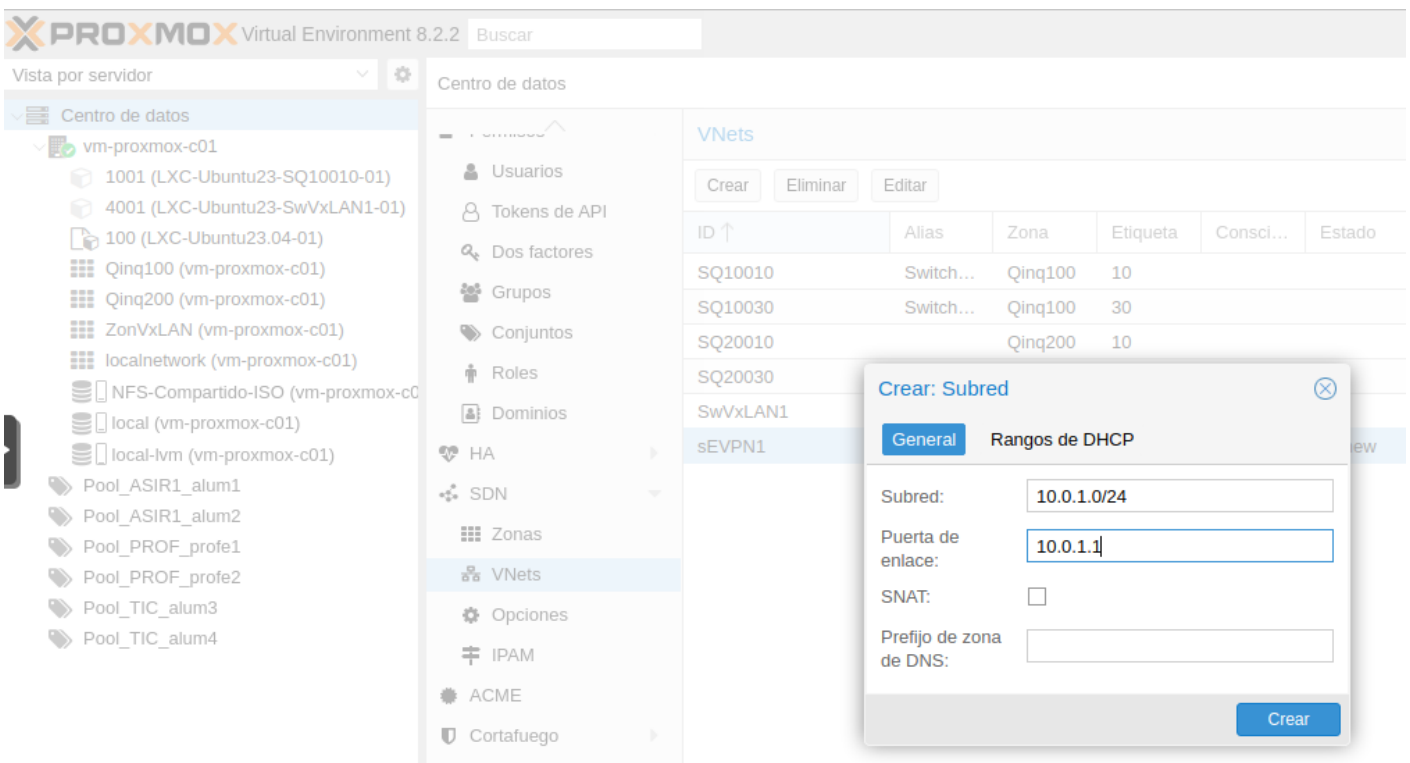


Imagen de elaboración propia: Creación de la subnet 10.0.1.0/24 (CC BY-NC-SA)

Aplica la configuración desde SDN y repetir el proceso en el otro nodo Proxmox.



¡ATENCIÓN! si no habilitamos en la Subnet SNAT, como es el caso anterior, las MV y contenedores no tendrán acceso al exterior de la VNet, es decir, no tendrán acceso a Internet.

Recordar que las MV o contenedores asociados a "sEVPN" deben tener un MTU de 1450 máximo y una IP estática (nodo c01 CT10.0.1.100 y nodo del c02 CT10.0.1.2) con puerta de enlace 10.0.1.1:

The screenshot shows the Proxmox VE interface for a container named '4001 (LXC-Ubuntu23-SwVxLAN1-01)'. The 'Red' tab is selected, showing a table of network interfaces:

ID ↑	Nombre	Puente	Cortafu...	Etiquet...	Dirección MAC	Dirección IP
net0	eth0	sEVPN1	Si		BC:24:11:05:...	10.0.1.100/24

The 'Editar: Dispositivo de red (veth)' modal window is open, showing the following configuration for the 'eth0' interface:

- Nombre: eth0
- Dirección MAC: BC:24:11:05:C8:A2
- Puente: sEVPN1
- Etiqueta VLAN: Ninguna VLAN
- Cortafuego:
- IPv4: Estático DHCP
- IPv4/CIDR: 10.0.1.100/24
- Puerta de enlace (IPv4): 10.0.1.1
- IPv6: Estático DHCP SLAAC
- IPv6/CIDR: Ninguna
- Puerta de enlace (IPv6):
- Desconectar:
- MTU: 1370
- Tasa límite (MB/s): unlimited

Buttons: Ayuda, Avanzado , Aceptar

Imagen de elaboración propia: Configuración de red del CT4001 con IP estática 10.0.1.100/24 y puerta de enlace 10.0.1.1

(CC BY-NC-SA)



PROXMOX Virtual Environment 8.2.2

Vista por servidor Ninguna etiqueta

- Centro de datos
 - vm-proxmox-c02
 - 1021 (LXC-Ubuntu23)
 - 1022 (LXC-Ubuntu23)
 - 2021 (LXC-Ubuntu23)
 - 2022 (LXC-Ubuntu23)
 - 4002 (LXC-Ubuntu23)
 - 100 (Plantilla-LXC-Ubuntu23)
 - Qinq100 (vm-proxmox-c02)
 - Qinq200 (vm-proxmox-c02)
 - ZonVxLAN (vm-proxmox-c02)
 - ZonaEVPN (vm-proxmox-c02)
 - localnetwork (vm-proxmox-c02)

- Resumen
- >_ Consola**
- Recursos
- Red
- DNS
- Opciones
- Historial de tareas
- Respaldo
- Replicación
- Snapshots

```

root@LXC-Ubuntu23-SxVLAN1-02:~# ping 10.0.1.100
PING 10.0.1.100 (10.0.1.100) 56(84) bytes of data.
64 bytes from 10.0.1.100: icmp_seq=1 ttl=64 time=1.42 ms
64 bytes from 10.0.1.100: icmp_seq=2 ttl=64 time=1.23 ms
64 bytes from 10.0.1.100: icmp_seq=3 ttl=64 time=1.21 ms
64 bytes from 10.0.1.100: icmp_seq=4 ttl=64 time=1.49 ms
64 bytes from 10.0.1.100: icmp_seq=5 ttl=64 time=1.21 ms
^Z
[2]+  Stopped                  ping 10.0.1.100
root@LXC-Ubuntu23-SxVLAN1-02:~#

```

PROXMOX Virtual Environment 8.2.2

Vista por servidor Ninguna etiqueta

- Centro de datos
 - vm-proxmox-c01
 - 1001 (LXC-Ubuntu23)
 - 4001 (LXC-Ubuntu23)
 - 100 (LXC-Ubuntu23)
 - Qinq100 (vm-proxmox-c01)
 - Qinq200 (vm-proxmox-c01)
 - ZonVxLAN (vm-proxmox-c01)
 - ZonaEVPN (vm-proxmox-c01)
 - localnetwork (vm-proxmox-c01)

- Resumen
- >_ Consola**
- Recursos
- Red
- DNS
- Opciones

```

64 bytes from 10.0.1.2: icmp_seq=103 ttl=64 time=1.13 ms
64 bytes from 10.0.1.2: icmp_seq=104 ttl=64 time=1.23 ms
64 bytes from 10.0.1.2: icmp_seq=105 ttl=64 time=1.06 ms
64 bytes from 10.0.1.2: icmp_seq=106 ttl=64 time=1.16 ms
64 bytes from 10.0.1.2: icmp_seq=107 ttl=64 time=1.30 ms
64 bytes from 10.0.1.2: icmp_seq=108 ttl=64 time=1.06 ms
64 bytes from 10.0.1.2: icmp_seq=109 ttl=64 time=1.21 ms
^C
--- 10.0.1.2 ping statistics ---
109 packets transmitted, 109 received, 0% packet loss, time 108210ms
rtt min/avg/max/mdev = 0.567/1.196/3.090/0.307 ms
root@LXC-Ubuntu23-SwVxLAN1-01:~#

```

Imagen de elaboración propia: Los dos contenedores, de diferentes nodos Proxmox, conectados y haciendo ping por EVPN

(CC BY-NC-SA)



5.- Licencia y autoría de este material

Materiales desarrollados inicialmente por **Daniel Cano Verdú (2024)** profesor de FP de la Junta de Andalucía y actualizados por el profesorado de la Junta de Andalucía bajo licencia **Creative Commons BY-NC-SA**.



Antes de cualquier uso leer detenidamente el siguiente [Aviso legal](#)