

Introducción a un sistema de almacenamiento con OpenMediaVault



ÍNDICE

Introducción a un sistema de almacenamiento con OpenMediaVault.....	1
Descripción.....	1
1. Objetivos que se quieren conseguir y se han conseguido.....	1
1.1 ¿Qué quiero conseguir con este proyecto?.....	1
1.2 ¿Que se ha conseguido?.....	1
2. Escenario necesario para la realización del proyecto.....	2
3. Fundamentos teóricos y conceptos.....	3
3.1 ¿Que es openmediavault?.....	3
3.2 Diferencia entre NAS y SAN.....	4
3.3 ¿Qué son los Sistemas de Archivos?.....	5
3.4 ¿Para que sirven los permisos de usuarios en la distribución linux?.....	8
3.5 ¿Que son los sistemas de bloque de ficheros en un disco duro?.....	9
3.6 Plugins Docker.....	10
3.7 LVM.....	11
4. Proceso de instalación y configuraciones.....	11
4.1 Instalación de openmediavault.....	11
4.2 Configuración y Gestión de usuarios.....	21
4.3 Creación de RAID con BTRFS.....	23
4.4 Creación de carpeta y compartir en la red.....	26
4.5 Configuración de iSCSI con LVM- Cliente Linux.....	33
4.6 Configuración de iSCSI con LVM – Cliente Windows.....	42
4.7 Configuración de Docker.....	50
4.8 Backups con rclone.....	56
5. Conclusiones y propuestas para seguir trabajando sobre el tema.....	66
6. Dificultades que se han encontrado.....	67
7. Bibliografía, enlaces, reseñas.....	67

Descripción

Mi proyecto se centra en la configuración de un servidor iSCSI utilizando la ISO de OpenMediaVault. Esta configuración nos permite compartir un disco duro ubicado en el servidor. Además, proporcionaremos una explicación detallada sobre la gestión de permisos de usuario, la compartición de carpetas en la red local y la configuración de plugins, configuración de un raid para la redundancia de datos. Por último, realizaremos una copia de seguridad de los datos más críticos de nuestro servidor en una ubicación externa para garantizar su replicación y seguridad.

1. Objetivos que se quieren conseguir y se han conseguido

1.1 ¿Qué quiero conseguir con este proyecto?

Mis objetivos principales con este proyecto (OMV), son a nivel educativo y empresarial:

Educación tecnológica accesible: Quiero proporcionar una herramienta educativa para personas sin conocimientos previos en tecnología. Mi proyecto está diseñado para que cualquier persona pueda seguirlo y replicarlo sin necesidad de buscar información adicional en internet. El objetivo es que adquieran los conocimientos básicos de informática, que, aunque no los convierta en expertos, sí les permitirá manejarse en su día a día con una base sólida de conocimientos técnicos.

Soporte a pequeñas empresas: Quiero ayudar a pequeñas empresas que no tienen recursos financieros suficientes para pagar servicios en la nube mensualmente. Aunque este proyecto requiere una inversión inicial en hardware, no es necesario adquirir equipos de última generación. Utilizamos un sistema operativo basado en Linux que consume muy pocos recursos, lo que permite gestionar servidores propios sin depender de plataformas de nubes privadas.

Es una manera de capacitar a personas y apoyar a pequeñas empresas para que sean autosuficientes en la gestión de sus recursos tecnológicos, reduciendo costos y dependencia de terceros.

1.2 ¿Que se ha conseguido?

Con OMV, se han logrado varios avances significativos en la gestión de un servidor propio en casa:

Gestión de usuarios y permisos: Se ha establecido un sistema para la gestión de usuarios, asignándoles permisos específicos para asegurar el control y la seguridad dentro del servidor.

Creación de RAID con BTRFS: Se ha implementado un RAID utilizando el sistema de archivos BTRFS, lo que garantiza la redundancia de datos y fortalece la fiabilidad del servidor.

Compartición de carpetas en red: Se han configurado servicios como Samba y NFS para permitir la compartición de carpetas en la red local, facilitando el acceso y la colaboración entre los usuarios.

Configuración de iSCSI: Se ha incorporado la configuración de iSCSI para la compartición de discos duros, mejorando la velocidad y la eficiencia del servidor.

Implementación de Kubernetes: Se ha explicado y configurado el plugins de Kubernetes, una tecnología crucial para la gestión de clústeres, permitiendo la administración de programas locales en la red propia de la empresa. Esto facilita la interacción de los empleados con las aplicaciones necesarias para su trabajo.

Copias de seguridad programadas: Se ha establecido un sistema para realizar y programar copias de seguridad del servidor, asegurando la protección de datos a largo plazo.

2. Escenario necesario para la realización del proyecto.

El escenario que vamos a utilizar para desarrollar este proyecto con OMV es el siguiente:

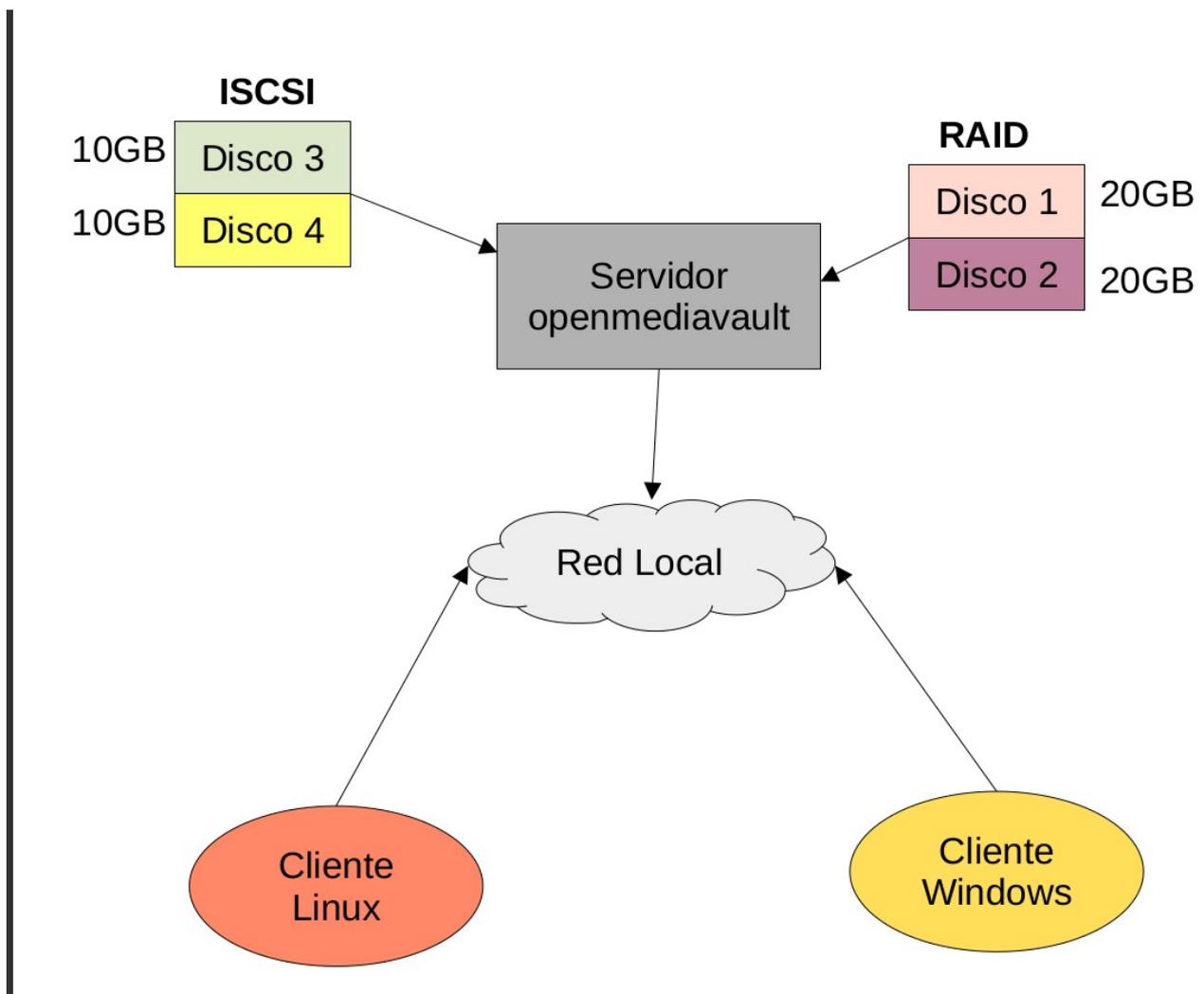
Máquina virtual como servidor: Utilizaremos una máquina virtual para realizar las funciones del servidor. Esta máquina virtual tendrá instalada la ISO de OpenMediaVault, con una capacidad de aproximadamente 20 GB. Esta configuración es suficiente y no requiere muchos plugins adicionales.

Discos duros para RAID: Asignaremos a la máquina virtual dos discos duros de 20 GB cada uno para la configuración de RAID con Btrfs. Esto permitirá la compartición de datos mediante carpetas por la red, mejorando la eficiencia y velocidad del servidor.

Discos duros para iSCSI: Asignaremos a la máquina virtual dos discos duros de 10 GB cada uno para la configuración de iSCSI con LVM. Esto permitirá la compartición de un disco para el cliente Windows y otro para el cliente Linux. La configuración de iSCSI facilitará la compartición de discos duros por la red, mejorando la eficiencia y velocidad del servidor.

Cliente Linux: Mi propia máquina funcionará como un cliente Linux para interactuar con el servidor.

Cliente Windows (Máquina virtual): También utilizaremos un cliente Windows, ya que la mayoría de los usuarios están acostumbrados a este sistema operativo y no suelen salir de su zona de confort.



3. Fundamentos teóricos y conceptos

3.1 ¿Que es openmediavault?

Es una distribución de Linux basada en Debian, diseñada para gestionar y crear dispositivos de almacenamiento en red. Lanzado en 2009, este proyecto surgió como una alternativa a FreeNAS. Volker Theile, su fundador, decidió crear OMV debido a las limitaciones de FreeNAS.

OMV se desarrolló para ofrecer una solución a los servidores NAS, utilizando una interfaz gráfica que es amigable para todos los usuarios y permite configuraciones sin necesidad de usar la línea de comandos. Con OMV, se pueden gestionar usuarios, compartir archivos, y configurar múltiples servicios y complementos, haciéndolo una opción muy atractiva por su ausencia de limitaciones. Con el tiempo, las versiones de este sistema operativo han mejorado, aumentando su funcionalidad y seguridad. Este proyecto ha ganado popularidad debido a su flexibilidad y capacidad de aplicación, convirtiéndose en uno de los más desarrollados y solicitados por los usuarios.

OMV permite la sincronización con la nube y el uso de herramientas de respaldo, entre otras características destacables. Además, es un sistema operativo gratuito y de código abierto, lo que lo hace accesible tanto para usuarios como para pequeñas empresas.

Cronologías de versiones:

Versión	Codename	Base Distro	Status	Date Released
0.2	Ix	Debian 6	EOL	Oct 2011
0.3	Omnious	Debian 6	EOL	Jul 2012
0.4	Fedaykin	Debian 6	EOL	Sep 2012
0.5	Sardoukar	Debian 6	EOL	Aug 2013
1.0	Kralizec	Debian 7	EOL	Sept 2014
2.0	Stoneburner	Debian 7	EOL	Jun 2015
3.0	Erasmus	Debian 8	EOL	Jun 2016
4.0	Arrakis	Debian 9	EOL	Apr 2018
5.0	Usul	Debian 10	EOL	Mar 2020
6.0	Shaitan	Debian 11	Oldstable	May 2022
7.0	Sandworm	Debian 12	Stable	Mar 2024

3.2 Diferencia entre NAS y SAN

3.2.1 NAS

Es una solución de almacenamiento a nivel de archivos que se conecta a una LAN y proporciona recursos de almacenamiento compartido a múltiples clientes o servidores. Los dispositivos NAS tienen protocolos de red estándar como Ethernet, TCP/IP, NFS o SMB/CIFS y tienen estructuras de datos redundantes para una alta confiabilidad. Los sistemas NAS son fáciles de administrar y ofrecen un enfoque simplificado para compartir archivos y almacenar datos.

El sistema NAS ahora aparece como un servidor de archivos especialmente conectado al protocolo de red estándar. Utiliza la red Ethernet e IP para iniciar sesión en los clientes y los niveles de archivo de acceso, vienen con sus propios sistemas operativos y sistemas de archivos, lo que les permite administrar el almacenamiento de archivos y realizar diversas tareas de administración de datos de forma independiente. El sistema de archivos NAS permite almacenar y compartir archivos entre dispositivos. Los clientes pueden acceder a los archivos almacenados en el NAS utilizando protocolos como NFS o SMB.

3.2.2 SAN

Es una red dedicada de alta velocidad que conecta servidores y dispositivos de almacenamiento para compartir recursos de almacenamiento de red. Una SAN normalmente utiliza tecnología Fibre

Channel solo para crear una red de almacenamiento. Trabaja independientemente de la red de área local y proporciona acceso a la memoria a nivel de bloque, lo que lo hace idóneo para aplicaciones que requieren acceso rápido a datos con baja latencia.

Utiliza una arquitectura de red de almacenamiento dedicada para proporcionar acceso directo, de alta velocidad y a nivel de bloque a los dispositivos de almacenamiento.

¿Que es Fibre Channel?

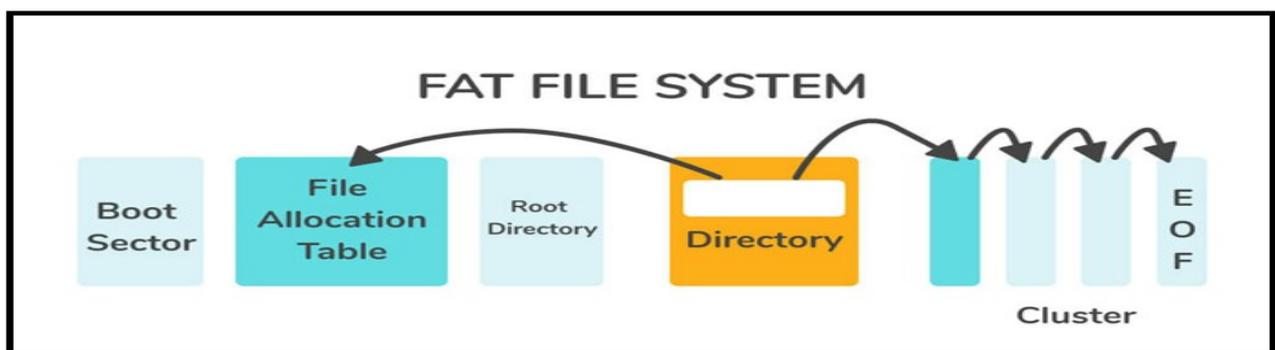
Es la tecnología más utilizada que conecta servidores y dispositivos de almacenamiento mediante conmutadores y controladores de almacenamiento dedicados. Las SAN generalmente se configuran con un controlador de almacenamiento que administra discos físicos y los presenta como discos lógicos a los servidores conectados. Los servidores acceden a estas unidades lógicas como si fueran dispositivos de almacenamiento conectados directamente, lo que permite la transferencia de datos de alta velocidad.

3.3 ¿Qué son los Sistemas de Archivos?

Los sistemas de archivos son utilizados por los sistemas operativos para organizar y almacenar datos en dispositivos de almacenamiento, como discos duros, unidades flash y tarjetas de memoria. Estos sistemas definen la estructura de los datos almacenados y proporcionan métodos para acceder, leer y escribir información de manera mas eficiente.

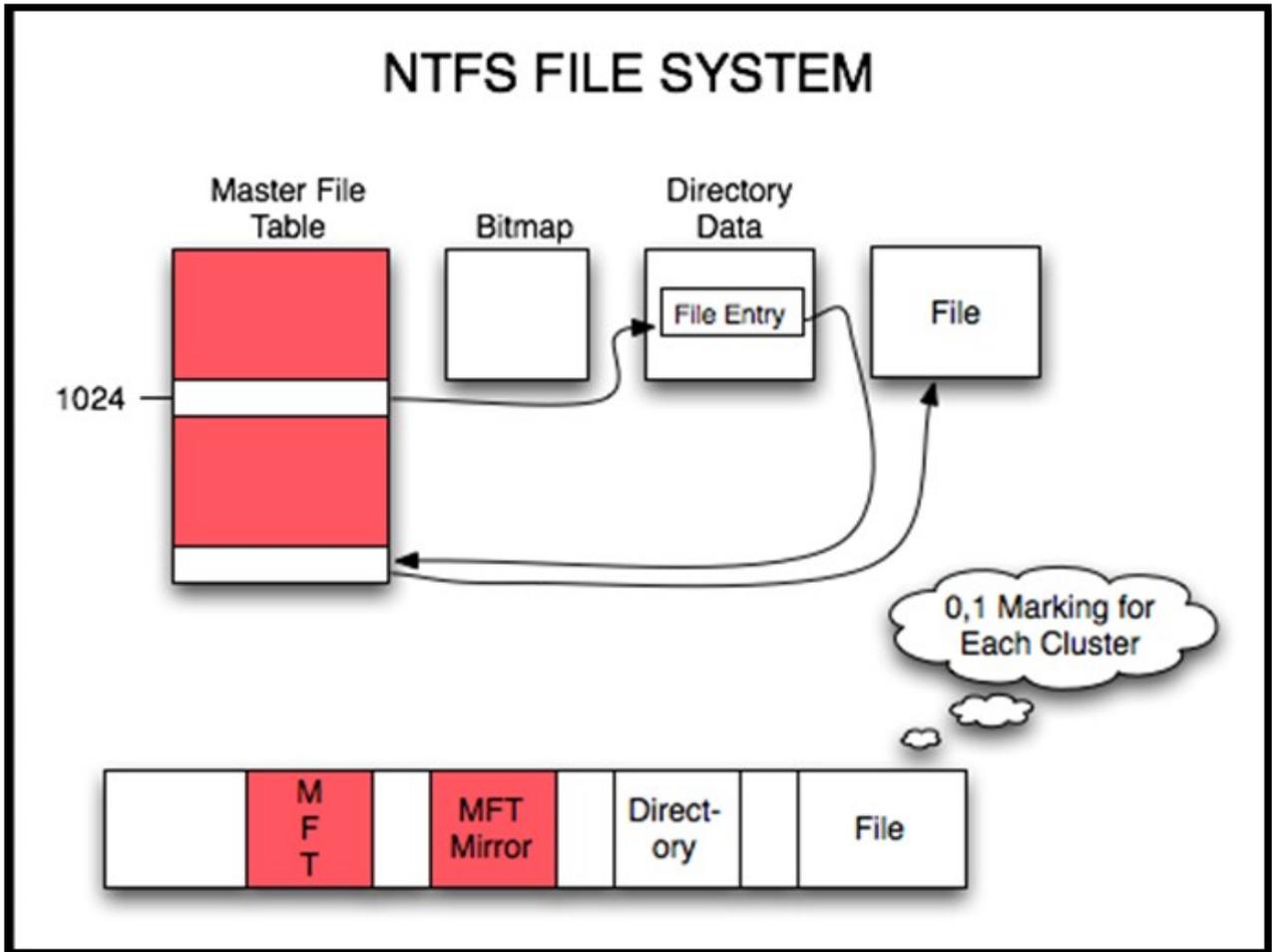
5.3.1 FAT

Es uno de los sistemas de archivos más antiguos y simples. Utiliza una tabla de asignación de archivos para almacenar la ubicación de los archivos en el disco. FAT se utiliza comúnmente en dispositivos USB, tarjetas de memoria y sistemas Windows.



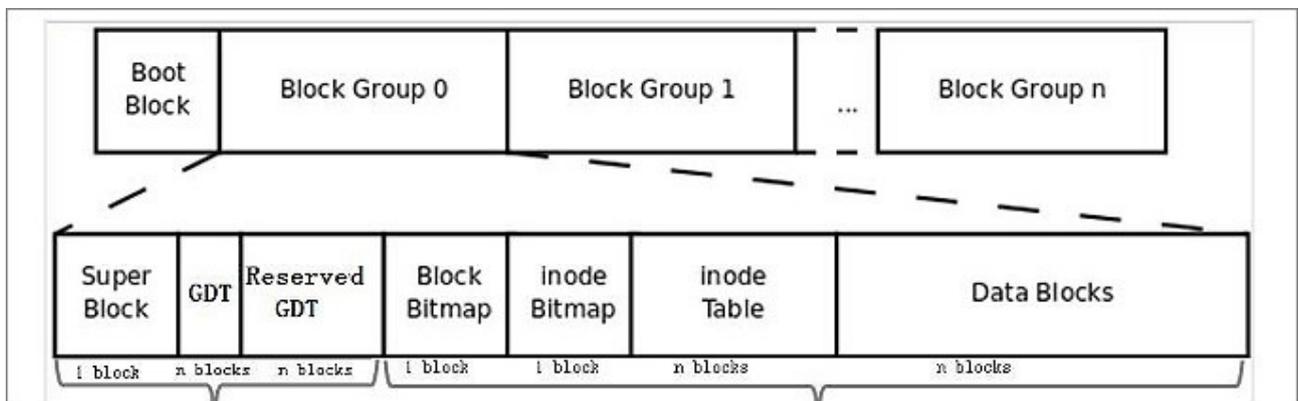
3.3.2 NTFS

Es un sistema de archivos desarrollado por Microsoft para su uso en sistemas Windows. Ofrece características avanzadas como permisos de archivo, compresión de archivos, cifrado y registro del sistema para mayor confiabilidad.



3.3.3 ext4

Es una versión mejorada del sistema de archivos ext3 utilizado en sistemas Linux. Ofrece mejoras en el rendimiento y la capacidad de almacenamiento, así como soporte para archivos más grandes y sistemas de archivos más grandes.



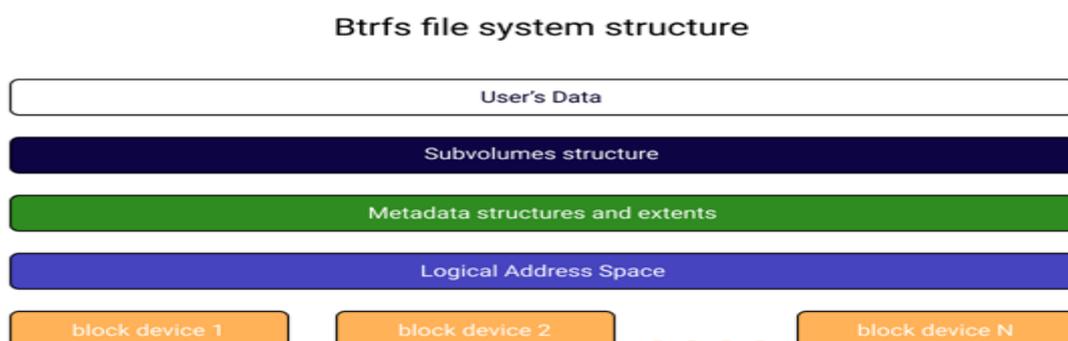
3.3.4 Btrfs

Es un sistema de archivos de código abierto diseñado para sistemas operativos Linux. Introducido como un proyecto experimental en 2007, se ha vuelto cada vez más popular debido a sus características avanzadas y su capacidad para manejar grandes volúmenes de datos.

Utiliza una estructura de árbol B para organizar y administrar los datos en el disco. Esta estructura ofrece ventajas significativas en términos de escalabilidad, rendimiento y confiabilidad. Algunas de las características clave de Btrfs incluyen:

- **Snapshots:** Permite la creación instantánea de instantáneas de los sistemas de archivos, lo que facilita la copia de seguridad y la restauración de datos.
- **Sub Volúmenes:** Permite la creación de sub volúmenes, que son como directorios separados con su propio espacio de almacenamiento y configuraciones de montaje.
- **Compresión:** Admite la compresión de datos en tiempo real, lo que puede ahorrar espacio en el disco y mejorar el rendimiento de lectura y escritura.
- **Checksums:** Utiliza checksums para detectar y corregir errores de datos, lo que mejora la integridad de los datos almacenados.
- **RAID:** Ofrece soporte integrado para la configuración de RAID, lo que permite la redundancia de datos y la protección contra fallos de disco.

Es una opción popular para sistemas de almacenamiento de datos en servidores Linux debido a su capacidad para manejar grandes cantidades de datos de manera eficiente y confiable. Es especialmente adecuado para entornos donde la integridad y la escalabilidad son críticas, como servidores de bases de datos y sistemas de almacenamiento en la nube.



3.4 ¿Para que sirven los permisos de usuarios en la distribución linux?

En la distribución linux la parte de los permisos de usuario es una de las cosas más importante que debemos de gestionar. Estos permisos hacen que el usuario pueda determinar unas series de acciones según lo que tenga asignados (editar archivos, borrar, actualizar...). Estos permisos se establecen a nivel de sistemas de archivos y se controla a nivel mediante tres tipos de permisos muy básicos :

- **Lectura (r - read):** Permite ver el contenido del archivo o listar el contenido de un directorio.
- **Escritura (w - write):** Permite modificar el contenido del archivo o hacer cambios dentro de un directorio (como crear o eliminar archivos).
- **Ejecución (x - execute):** Permite ejecutar un archivo (si es un programa o script) o acceder al contenido de un directorio.

Después tenemos tres categoría de usuarios:

- **Propietario (Owner):** El usuario que es dueño del archivo o directorio. El propietario puede ser cambiado usando el comando `chown`.
- **Grupo (Group):** Un conjunto de usuarios que comparten ciertos permisos sobre el archivo o directorio. El grupo asociado a un archivo se puede cambiar con `chgrp`.
- **Otros (Others):** Todos los demás usuarios que no son ni el propietario ni parte del grupo asociado al archivo o directorio.

Estos permisos se representa de este modo “-rwxr-xr-- “ :

- **Propietario:** Los primeros tres caracteres (rwx) indican los permisos del propietario.
- **Grupo:** Los siguientes tres caracteres (r-x) indican los permisos del grupo.
- **Otros:** Los últimos tres caracteres (r--) indican los permisos de los demás usuarios.

Comando básicos que deberíamos saber para la gestión de permisos:

- **ls -l:** Lista archivos y directorios junto con sus permisos.
- **chmod:** Cambia los permisos de archivos y directorios.
- **chown:** Cambia el propietario de archivos y directorios.
- **chgrp:** Cambia el grupo asociado a archivos y directorios.

Es muy importante que estos conceptos nos quede claro ya que no vamos a realizar una configuración a través de una terminal, pero si lo vamos a realizar por una interfaz gráfica, el cual es parecido y este tema de permisos nos saldrá a la hora de la creación de un usuario, carpetas...

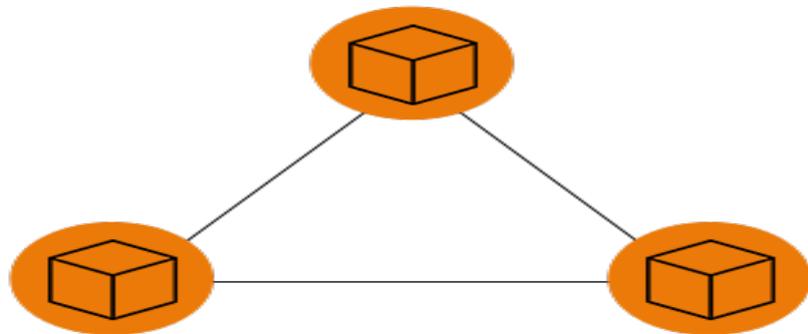
3.5 ¿Que son los sistemas de bloque de ficheros en un disco duro?

El sistema de archivos de bloques y el sistema de archivos son dos conceptos clave en los sistemas operativos para administrar discos duros y almacenamiento. Aquí te detallamos qué son y cómo funcionan.

3.5.1 Sistema de bloques

El sistema de bloques divide el disco duro en bloques de tamaño fijo (por ejemplo, 512 bytes, 4 KB...). Cada bloque tiene una dirección única y es la unidad básica de almacenamiento. Este enfoque es fundamental para los sistemas de archivos y bases de datos que requieren acceso directo a unidades de almacenamiento para realizar operaciones de lectura y escritura de manera eficiente

- **Acceso directo:** permiten el acceso directo a bloques individuales, lo cual es fundamental para aplicaciones que requieren alto rendimiento y baja latencia.
- **Flexibilidad:** los bloques se pueden administrar de forma independiente, lo que permite que el sistema de archivos organice y utilice el almacenamiento según sea necesario.
- **Versatilidad:** Se pueden utilizar en sistemas de almacenamiento local y sistemas de almacenamiento en red.



3.5.2 Sistema de archivos

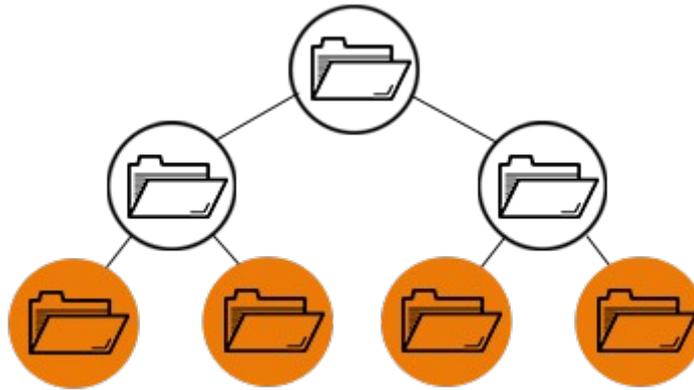
Es la capa superior que organiza y administra cómo se almacenan y recuperan los datos en el almacenamiento en bloque. Los sistemas de archivos convierten bloques de datos en estructuras lógicas más manejables, como archivos y directorios.

Estructura: Organiza los datos en una estructura jerárquica de archivos y directorios.

Metadatos: Mantienen información sobre archivos y directorios, como permisos, fechas de modificación, tamaño...

Facilidad de su uso: Facilitan a los usuarios y aplicaciones la creación, lectura, modificación y eliminación de archivos.

El sistema de bloques es esencial para almacenar datos a nivel físico, mientras que el sistema de archivos proporciona la estructura lógica para gestionar y organizar los datos. Los dos trabajan juntos para proporcionar un sistema de almacenamiento eficiente, organizado y seguro. La explicación de cada uno de ellos no los podemos encontrar en el apartado : **3.3 ¿Qué son los Sistemas de Archivos?**



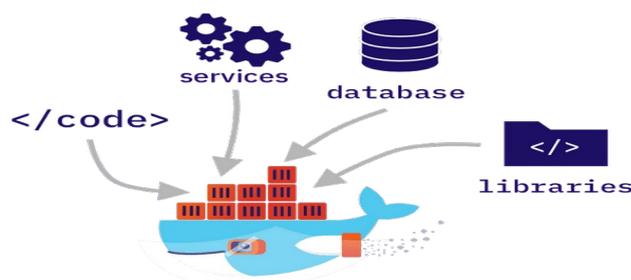
3.6 Plugins Docker.

Docker es una herramienta de código abierto que simplifica la creación, el despliegue y la ejecución de aplicaciones en contenedores.

Estos contenedores son como paquetes de software autónomos que contienen todo lo necesario para que una aplicación funcione de manera consistente, incluyendo bibliotecas, dependencias y configuraciones.

Docker permite crear estos contenedores de aplicaciones junto con su entorno de ejecución, lo que garantiza que la aplicación se comporte de la misma manera en cualquier lugar donde se ejecute Docker, ya sea en entornos de desarrollo, pruebas o producción.

Algunas de las ventajas de utilizar Docker son su capacidad de portabilidad, ya que los contenedores pueden ejecutarse en cualquier sistema compatible con Docker independientemente de su infraestructura, y la eficiencia en el uso de recursos, dado que los contenedores comparten recursos del sistema operativo sin redundancias. Esto hace que Docker sea una herramienta popular para desarrolladores y administradores de sistemas que desean crear y gestionar aplicaciones de manera más eficiente y confiable.



3.7 LVM

Es una tecnología de gestión de almacenamiento utilizada en sistemas operativos basados en Linux. Proporciona una capa de abstracción entre el disco duro físico y el sistema de archivos, proporcionando mayor flexibilidad y eficiencia en la gestión de los recursos de almacenamiento. En lugar de manejar discos duros físicos individuales, LVM organiza el espacio de almacenamiento en bloques lógicos que se pueden cambiar de tamaño, mover y agrupar según sea necesario sin interrumpir el acceso a los datos. Esto le permite realizar tareas como crear particiones de tamaño dinámico, mover datos entre discos y crear instantáneas de disco en tiempo real. LVM incluye varios elementos clave:

Volúmenes físicos (PV): estos son los discos duros físicos o particiones de disco que se utilizan como componentes de almacenamiento principal.

Grupo de volúmenes (VG): agrupa uno o más volúmenes físicos en un espacio de almacenamiento lógico. Un grupo de volúmenes es un grupo de almacenamiento que se puede asignar a bloques lógicos.

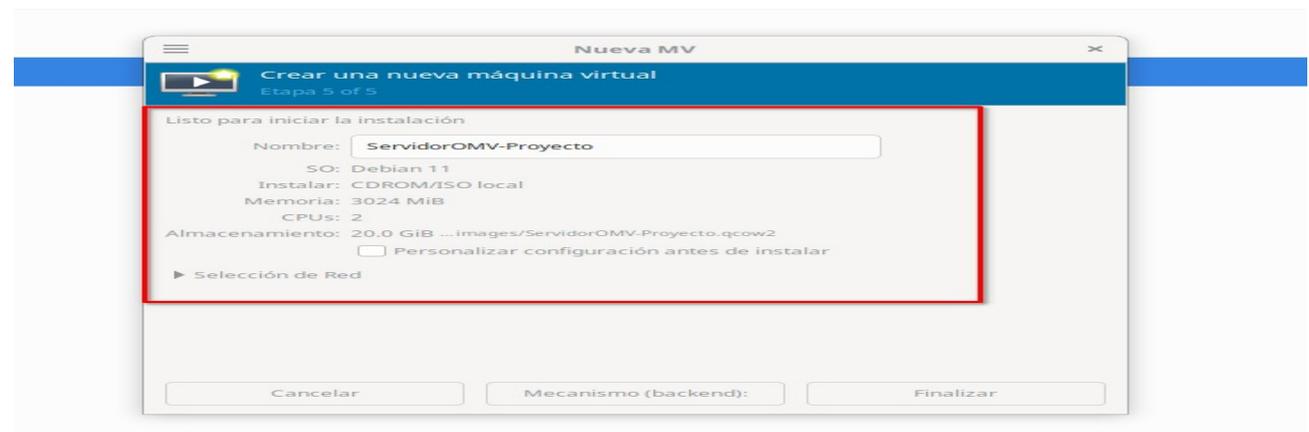
Volúmenes lógicos (LV): son unidades de almacenamiento lógico creadas en grupos de volúmenes. Los volúmenes lógicos actúan como discos duros independientes y se pueden formatear utilizando un sistema de archivos de almacenamiento de datos.

Proporciona varios beneficios importantes, como flexibilidad en la administración del almacenamiento, la capacidad de cambiar el tamaño de las unidades sobre la marcha (sin reiniciar el sistema) y la capacidad de implementar almacenamiento avanzado como instantáneas y migración de datos. entre unidades sin interferir con el acceso a ellas.

4. Proceso de instalación y configuraciones

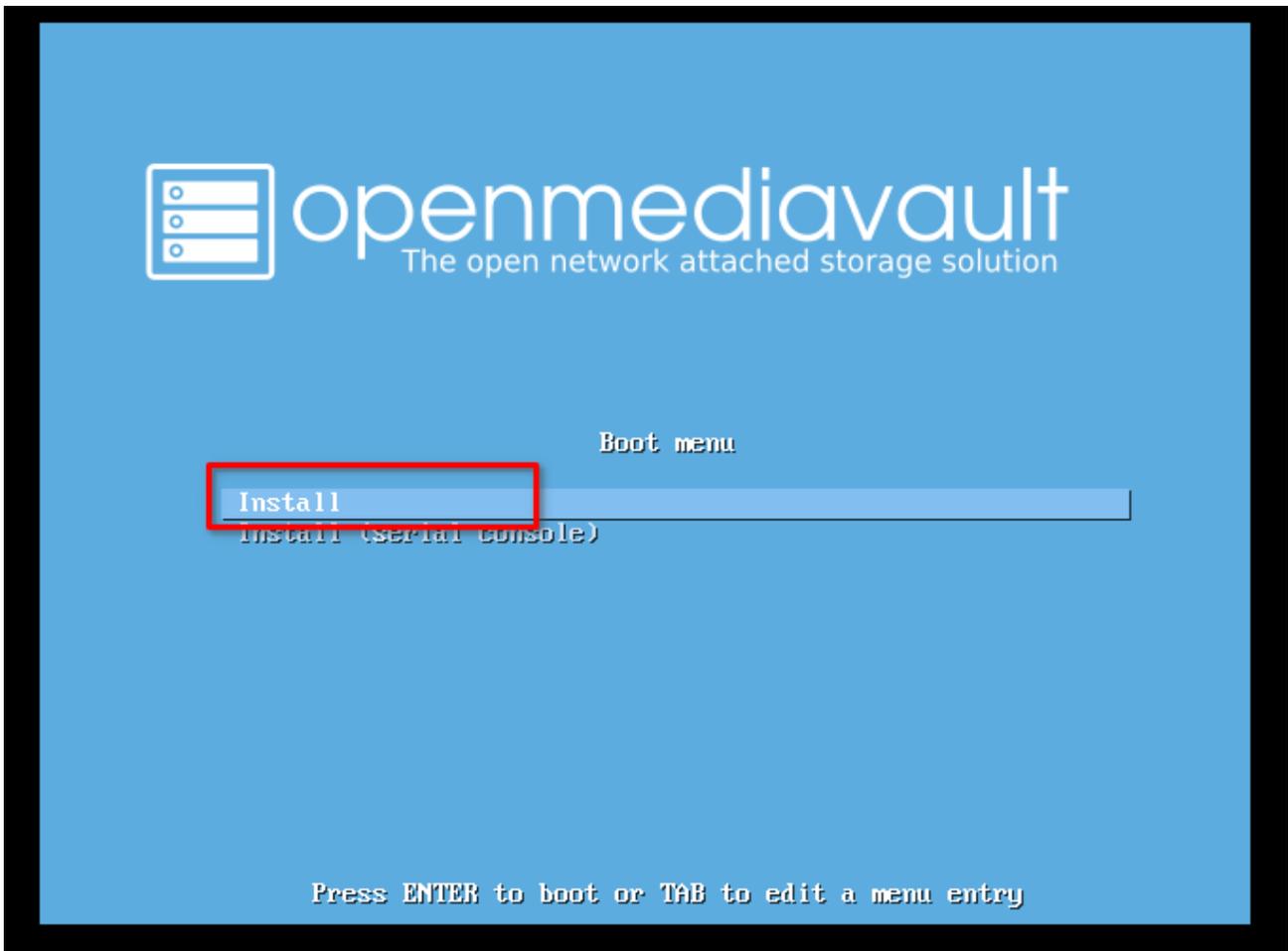
4.1 Instalación de openmediavault.

Primero creamos una maquina virtual con las siguientes características:

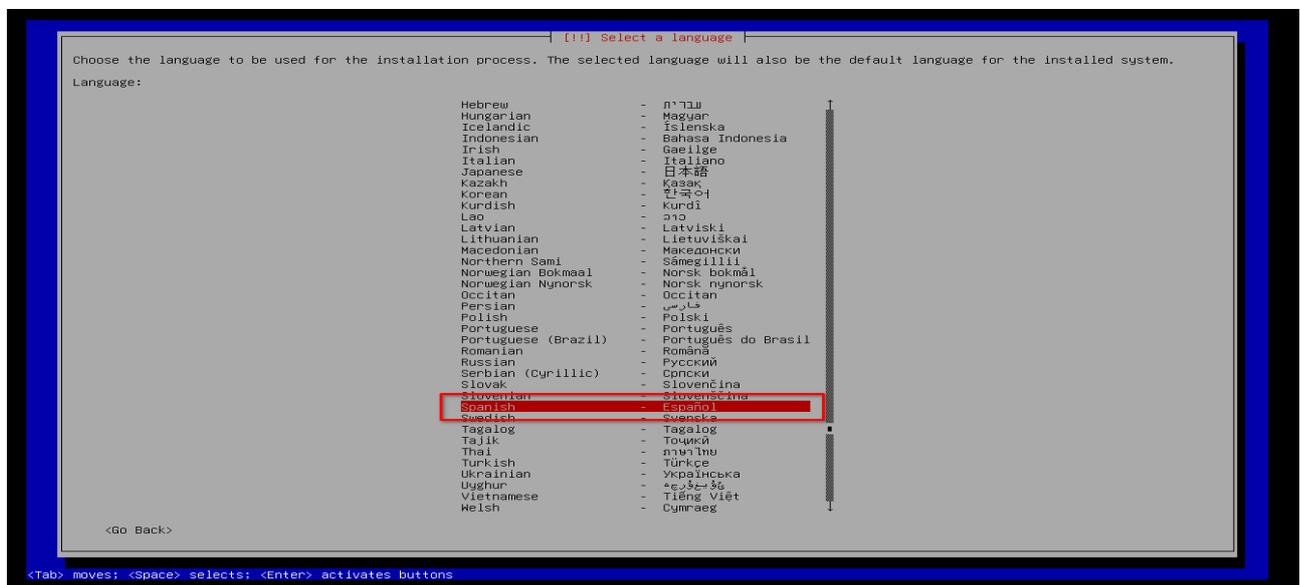


Una vez que tenemos la maquina configurada con estas características, vamos a proceder a la instalación de openmediavault. La configuración de la maquina es al gusto del cliente y las necesidades que el tenga.

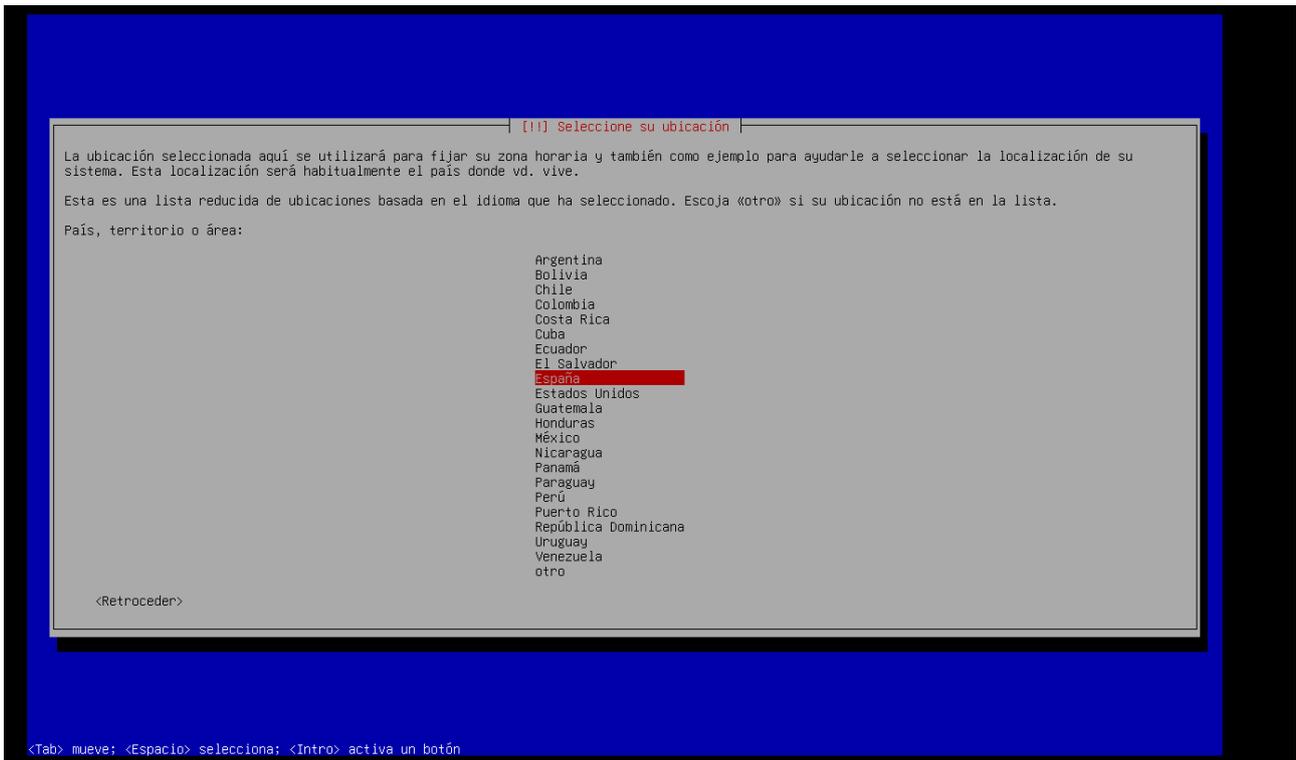
Seleccionamos instalar.



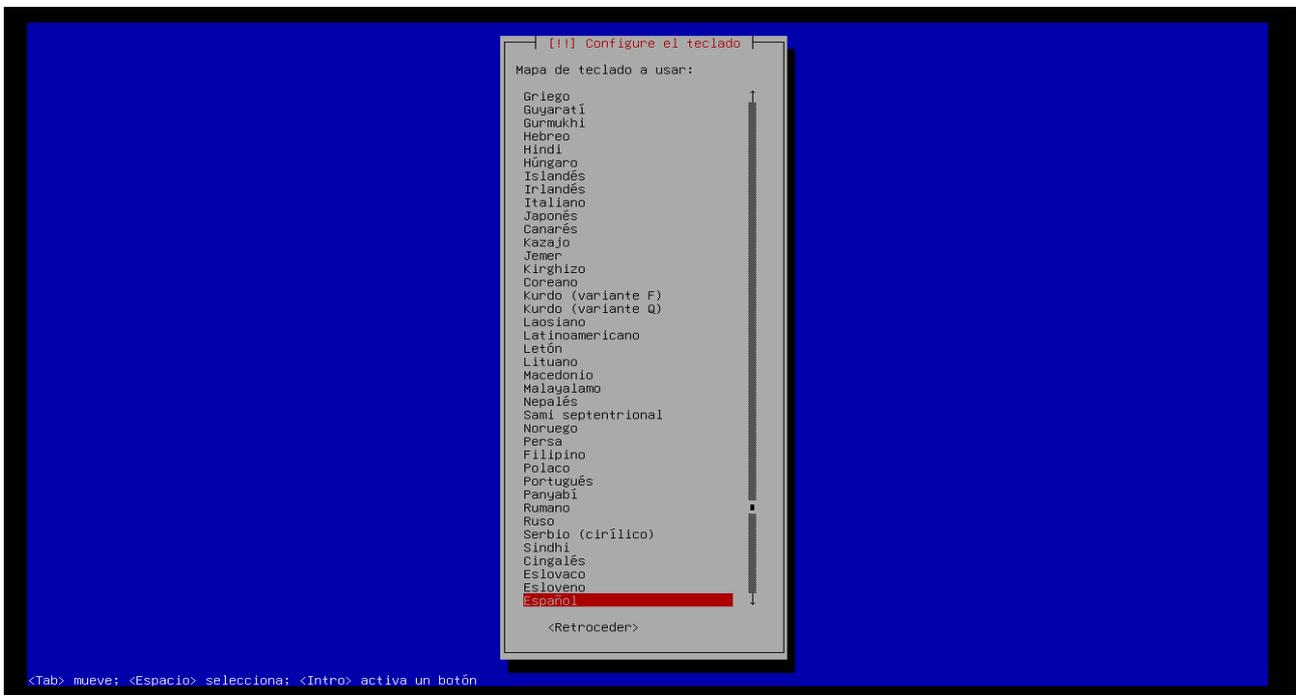
- Elegimos el idioma.



- Elegimos la ubicación del territorio actual.



- Configuramos el idioma de nuestro teclado :



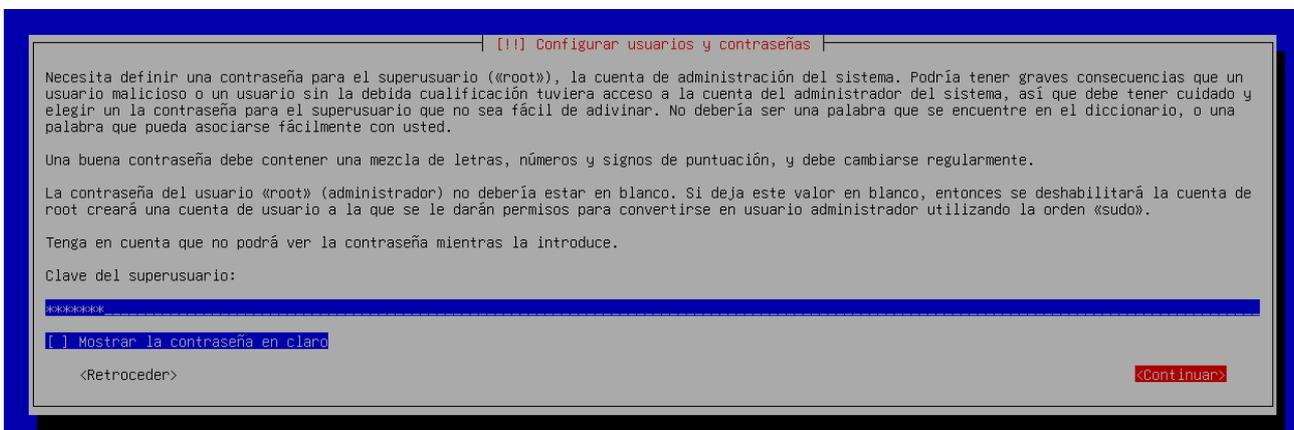
Una vez finalizado esta configuración previa a continuación nos pedirá el nombre de nuestro equipo, Lo podemos dejar tal y como esta o poner el nombre que mas nos guste. En mi caso lo dejara tal y como esta.



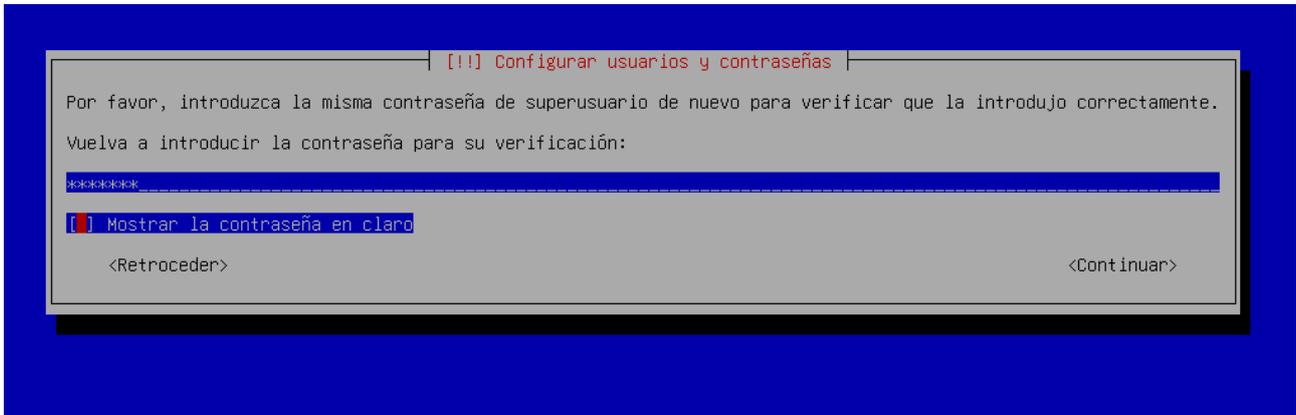
En el siguiente paso el instalador nos pedirá un dominio, por defecto sale local, como estamos configurando nuestro servidor para una red local lo dejamos en local. El nombre que obtendrá nuestra maquina sera openmediavault.local.



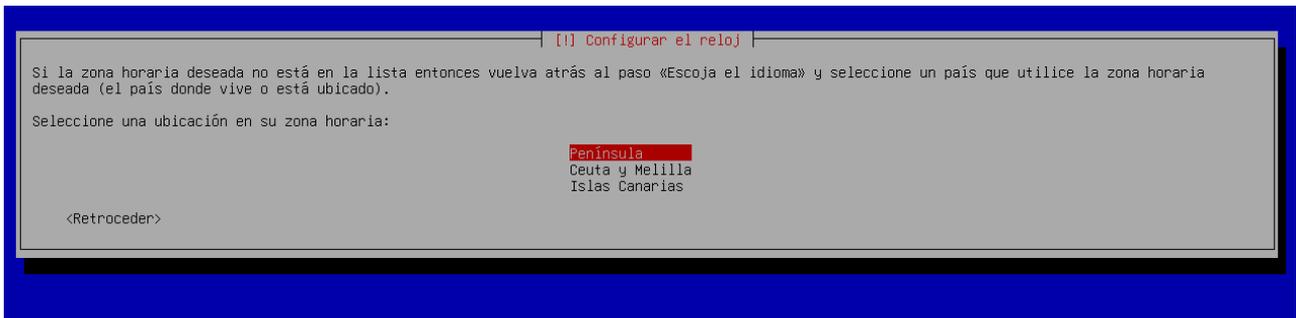
El siguiente paso será configurar la contraseña para el usuario administrador (root). Aunque esta contraseña no se utiliza para acceder a la interfaz web de administración, es esencial para iniciar sesión localmente con privilegios máximos.



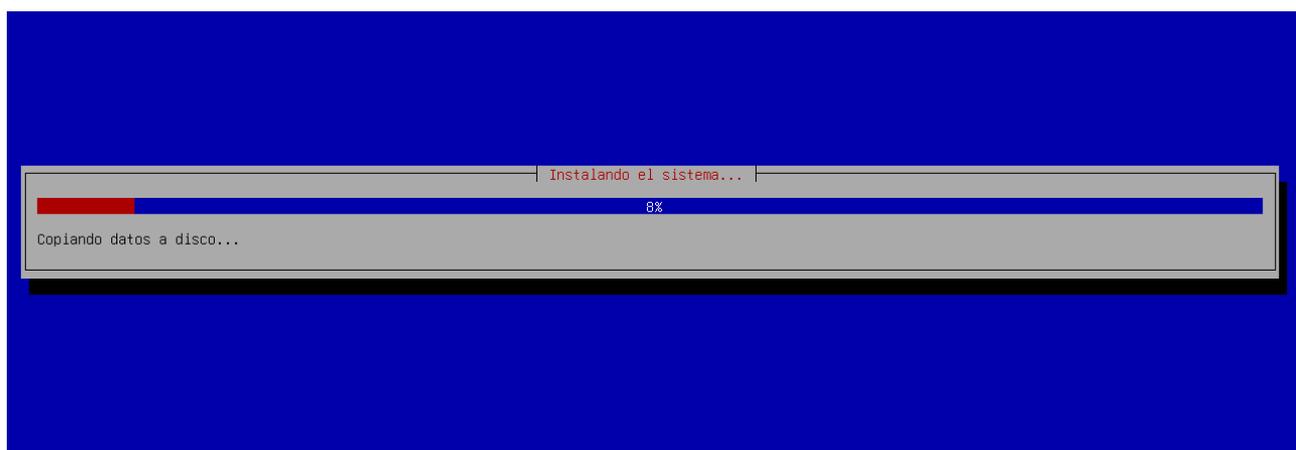
Debemos de repetir la contraseña que hemos puesto anteriormente para asegurarnos de que coincidan.



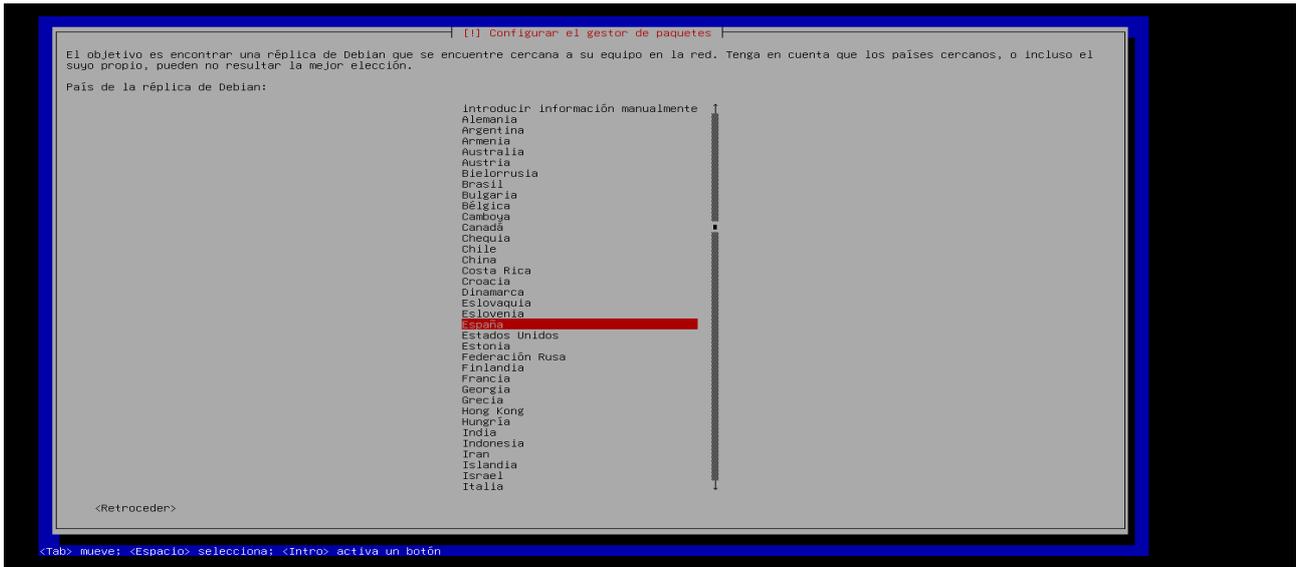
Elegimos a continuación la zona horaria:



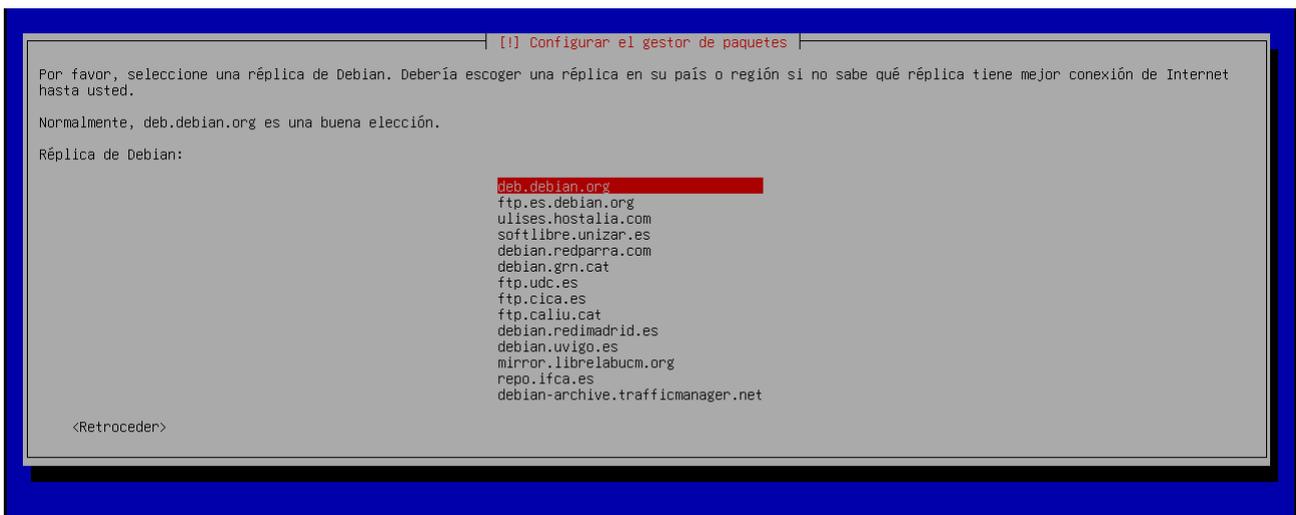
Una vez realizado las configuraciones previas, el sistema operativo ya es listo para realizar la copia de archivos al disco duro y comenzar con su propia instalación.



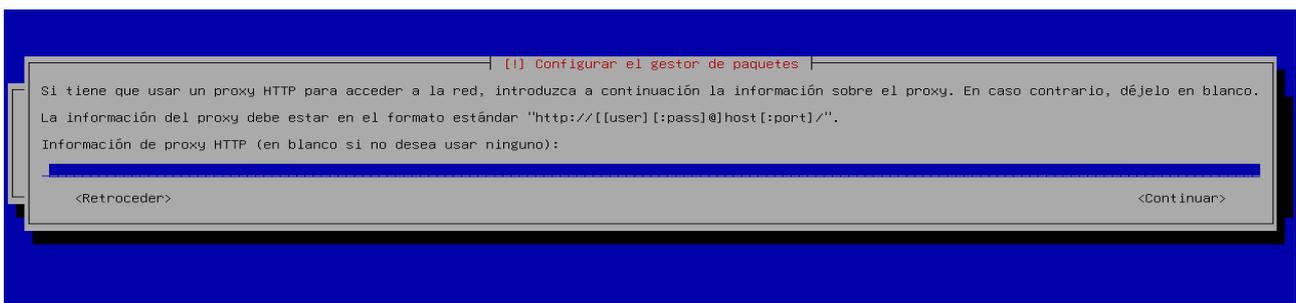
A continuación, debemos seleccionar el origen desde donde el administrador de paquetes descargará el software. Es recomendable elegir un servidor que esté cercano a nuestra ubicación geográfica. Por defecto elijo España.



Después, seleccionaremos el mirror de Debian que utilizará nuestro equipo. Es preferible elegir uno soportado por nuestro proveedor de Internet, pero si no disponemos de esta información, podemos optar por cualquier otro disponible.

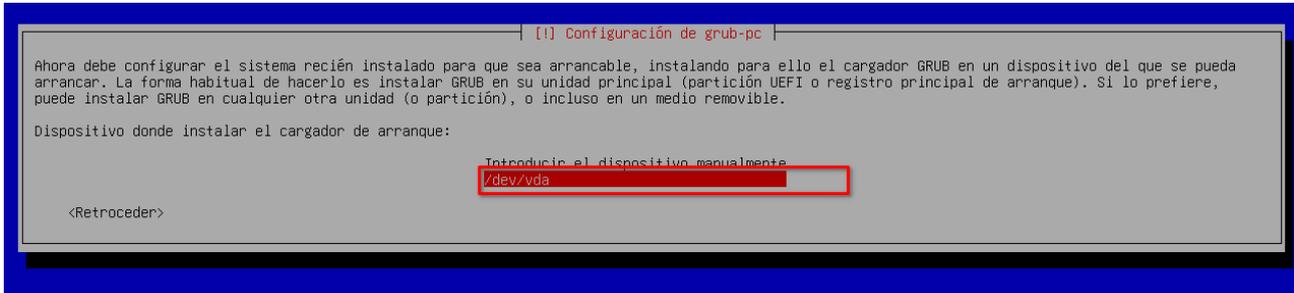


En la siguiente, proporcionaremos la información de nuestro servidor proxy, si disponemos de uno. Como no dispongo de un servidor proxy le doy intro y continuamos.

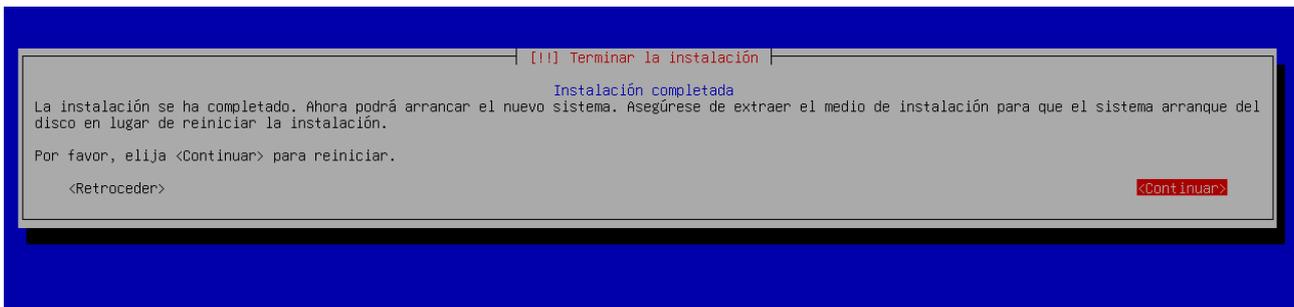


A partir de aquí, se completará la descarga de paquetes y comenzará la configuración del gestor de arranque. Debemos especificar el volumen en el que se instalará, que generalmente será el mismo disco donde instalamos el sistema. También tienes la opción de seleccionar el disco manualmente.

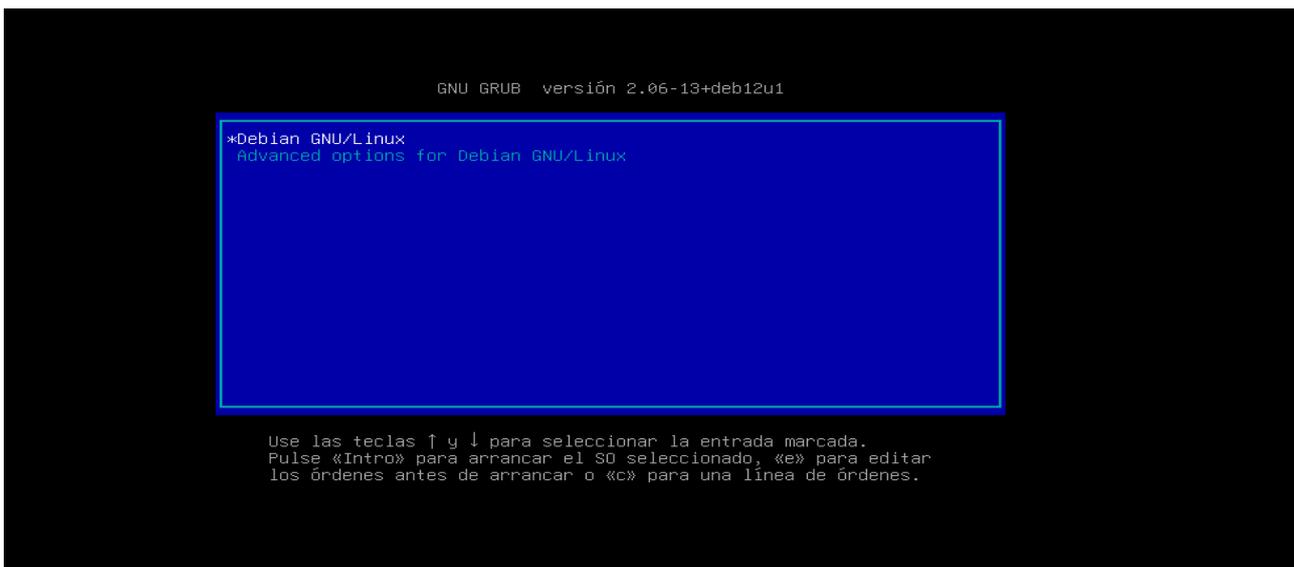
Si seguiste la recomendación inicial, este paso será muy sencillo, ya que solo tendrás un disco en el equipo.



Le damos a “Continuar “ para reiniciar el servidor.



Una vez reiniciado, aquí tenemos nuestro primer arranque donde podemos observar el grub.



Una vez reiniciado y arrancado nuestro sistema operativo, podemos observar que nos marca la interfaz de red con la ip a la cual deberemos acceder por internet y también nos dan las credenciales una vez estemos en el panel de inicio de sesión.

```
openmediavault 7.0-32 (Sandworm) openmediavault tty1
Copyright (C) 2009-2024 by Volker Theile. All rights reserved.

To manage the system visit the openmediavault workbench:
enp1s0: 192.168.122.227

By default the workbench administrator account has the
username 'admin' and password 'openmediavault'.
It is recommended that you change the password for this account
within the workbench or using the 'omv-firstaid' CLI command.

For more information regarding this appliance, please visit the
web site: https://www.openmediavault.org

Hint: Num Lock on

openmediavault login: _
```

Para acceder debemos de poner:

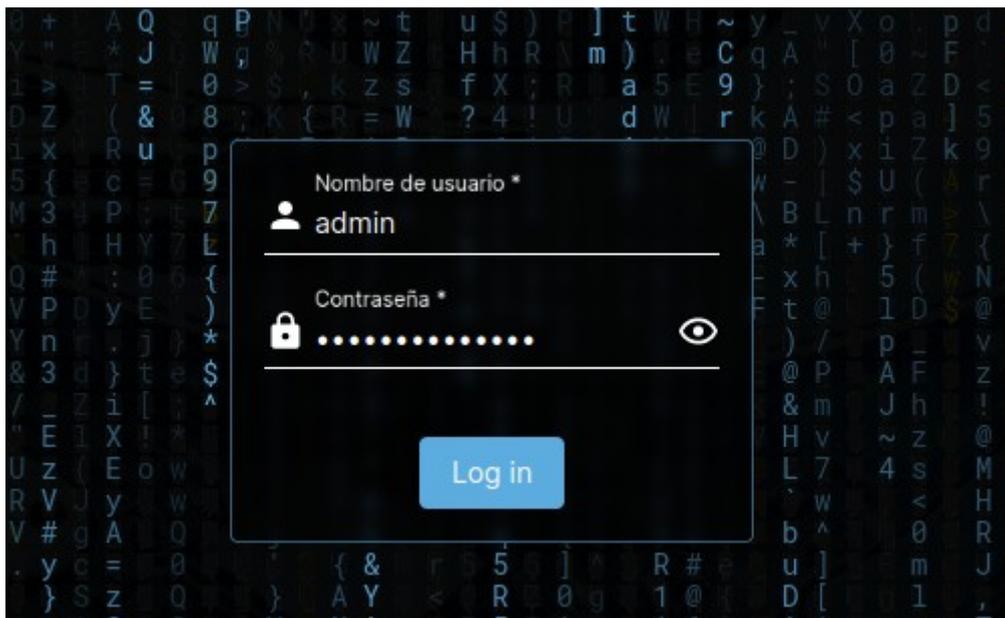
<http://192.168.122.227>



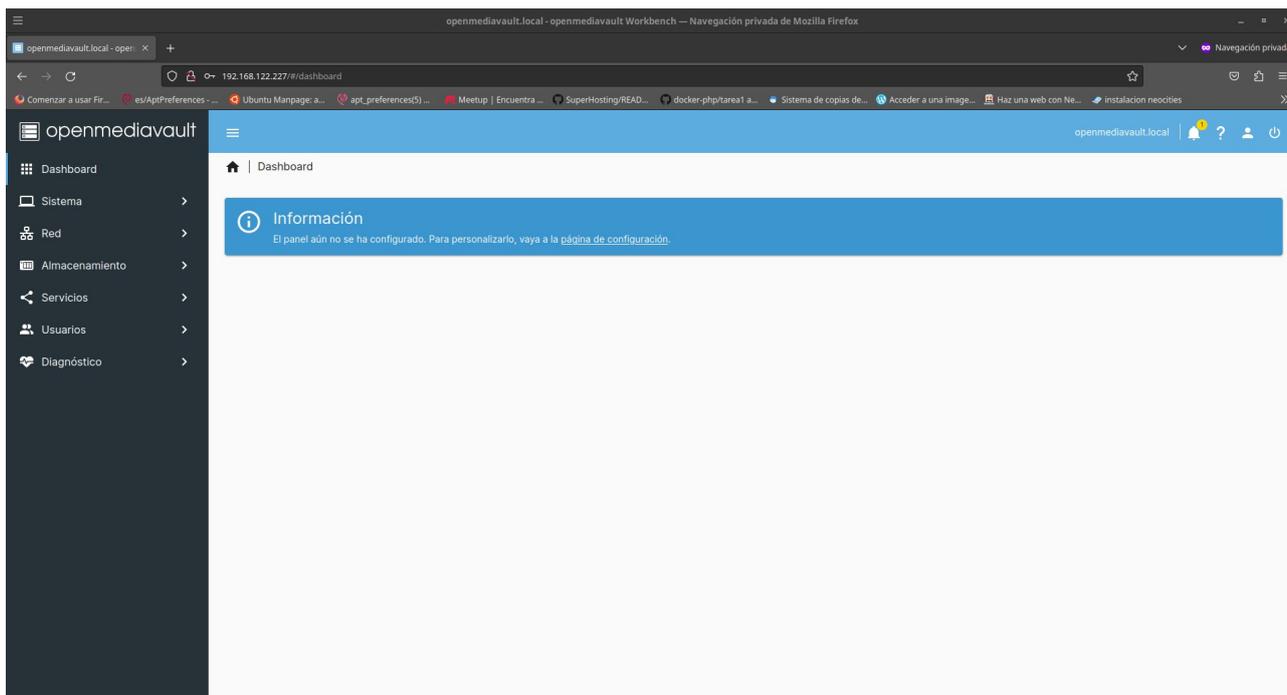
Credenciales:

Usuario: admin

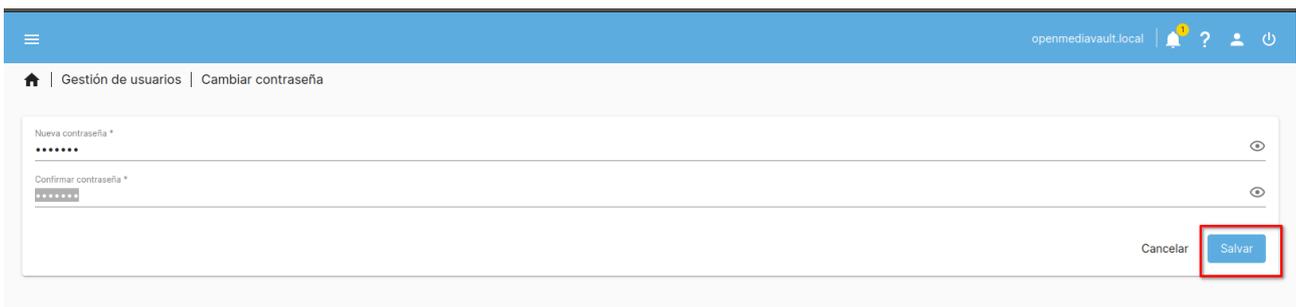
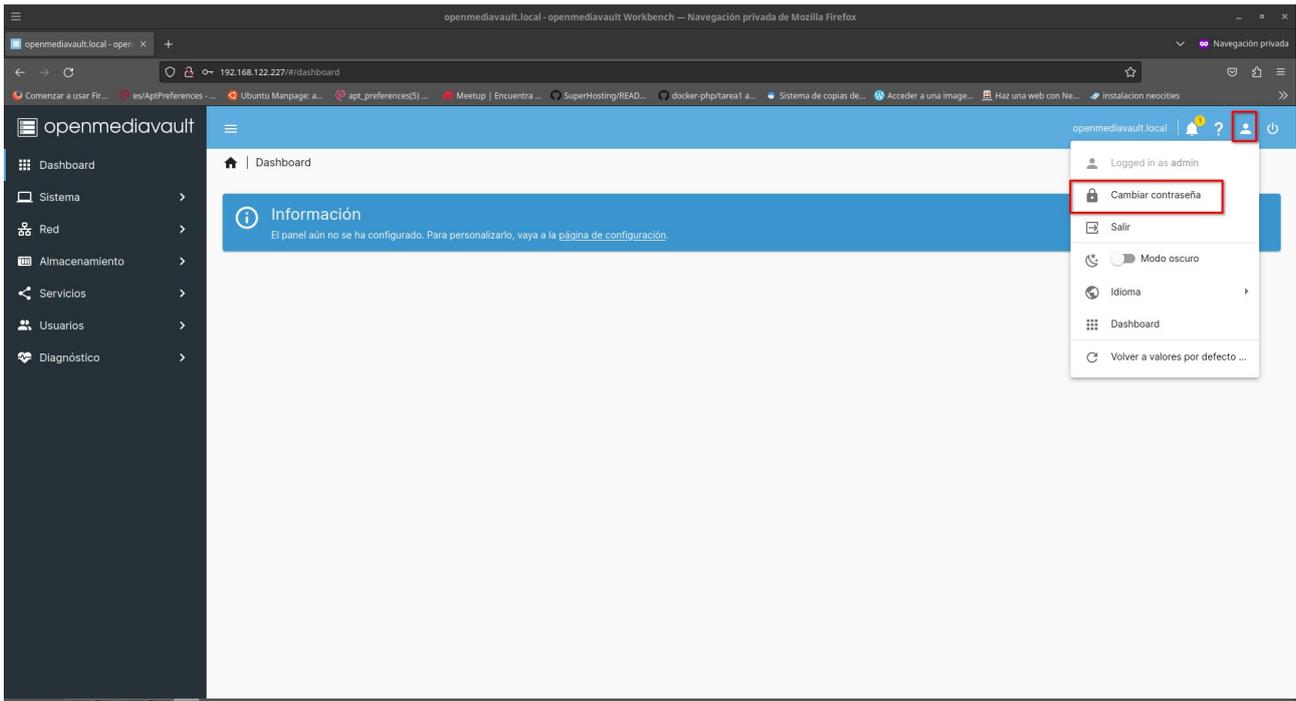
Contraseña: openmediavault



Listo para el comienzo de nuestra configuración.



Una vez que hemos ingresado en el panel de control gráfico, vamos a realizar el cambio de contraseña de para poner una contraseña mas amigable y que nos resulte útil para recordad.



Una vez modificada la contraseña podemos regresar al panel de inicio de sesión y volvemos a probar pero con la nueva contraseña:

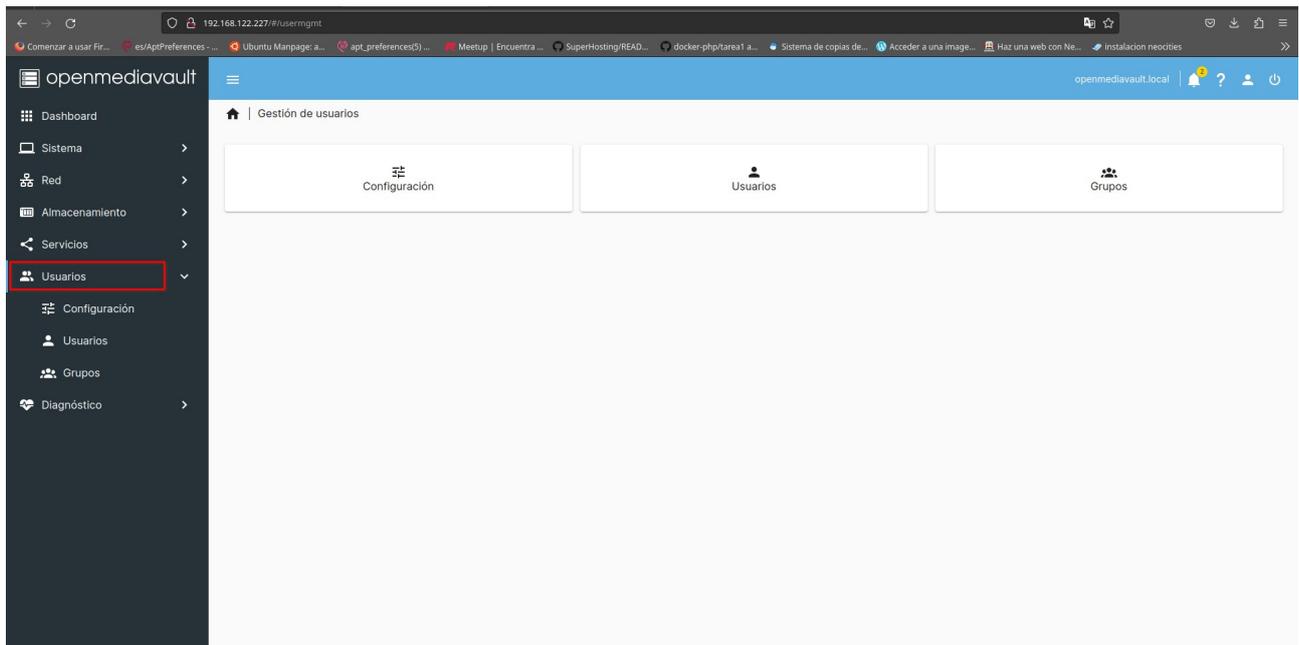


4.2 Configuración y Gestión de usuarios.

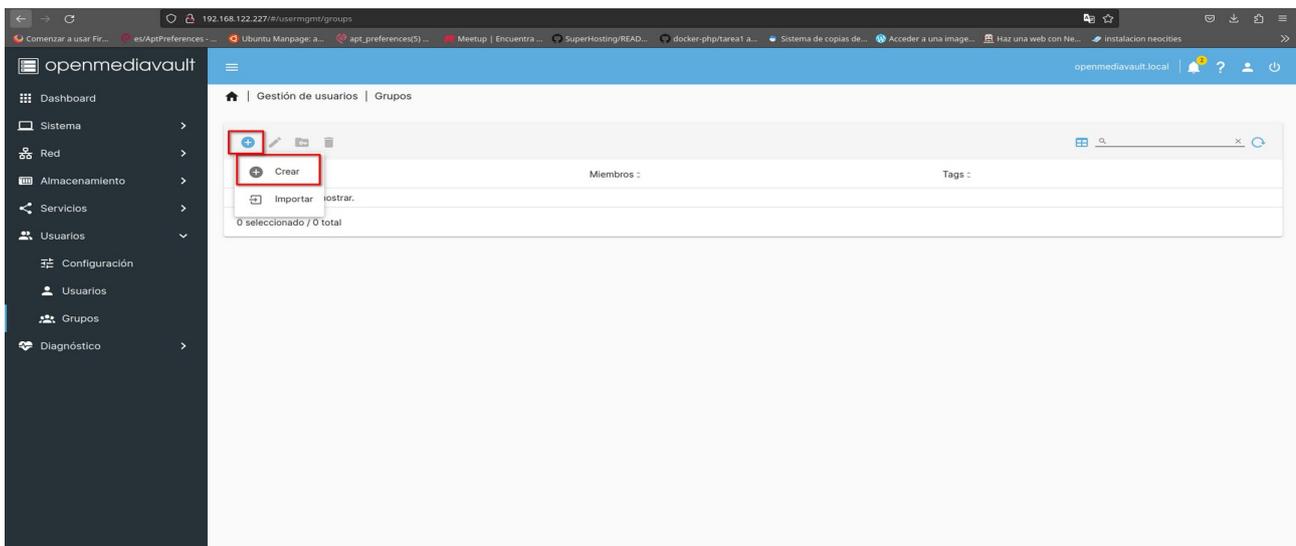
Para la configuración de los usuarios, es crucial tener una comprensión clara de los permisos y roles que deseamos asignar a estos usuarios en términos de su interacción con el servidor. Es especialmente importante que el administrador mantenga su rol como líder y jefe, sin perder autoridad frente a otros usuarios.

Para lograr esto, llevamos a cabo los siguientes pasos:

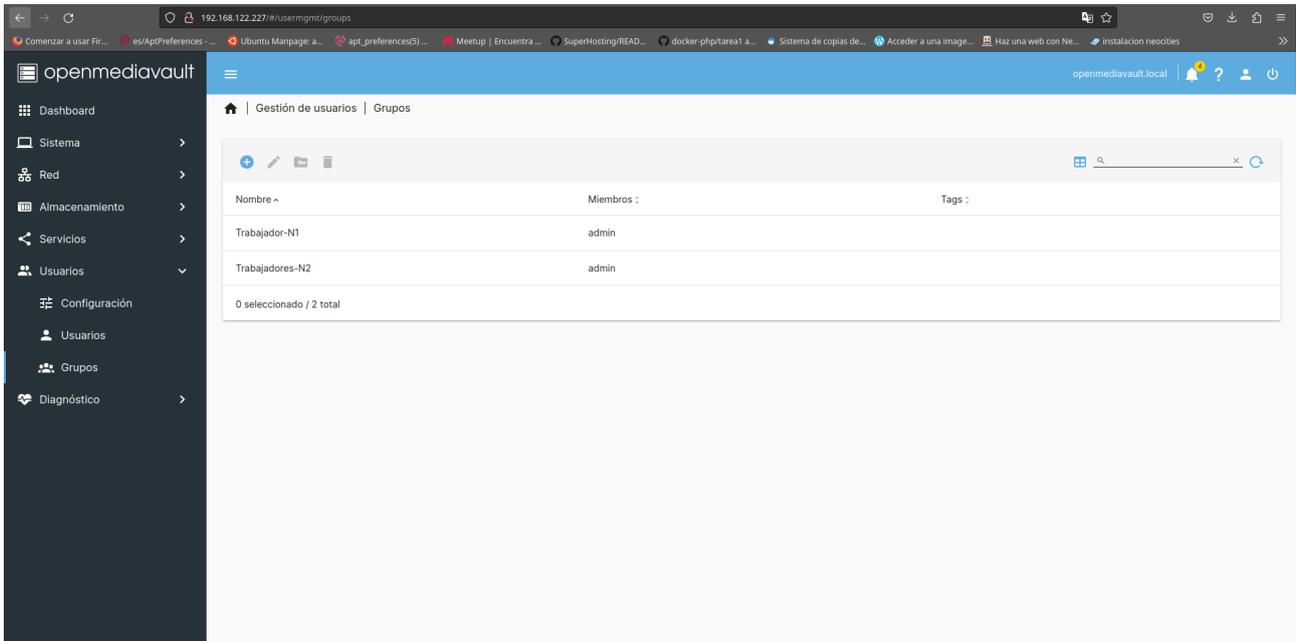
Nos dirigimos al apartado de usuarios, donde podemos configurar tanto a los usuarios como a los grupos en los que estos usuarios estarán incluidos.



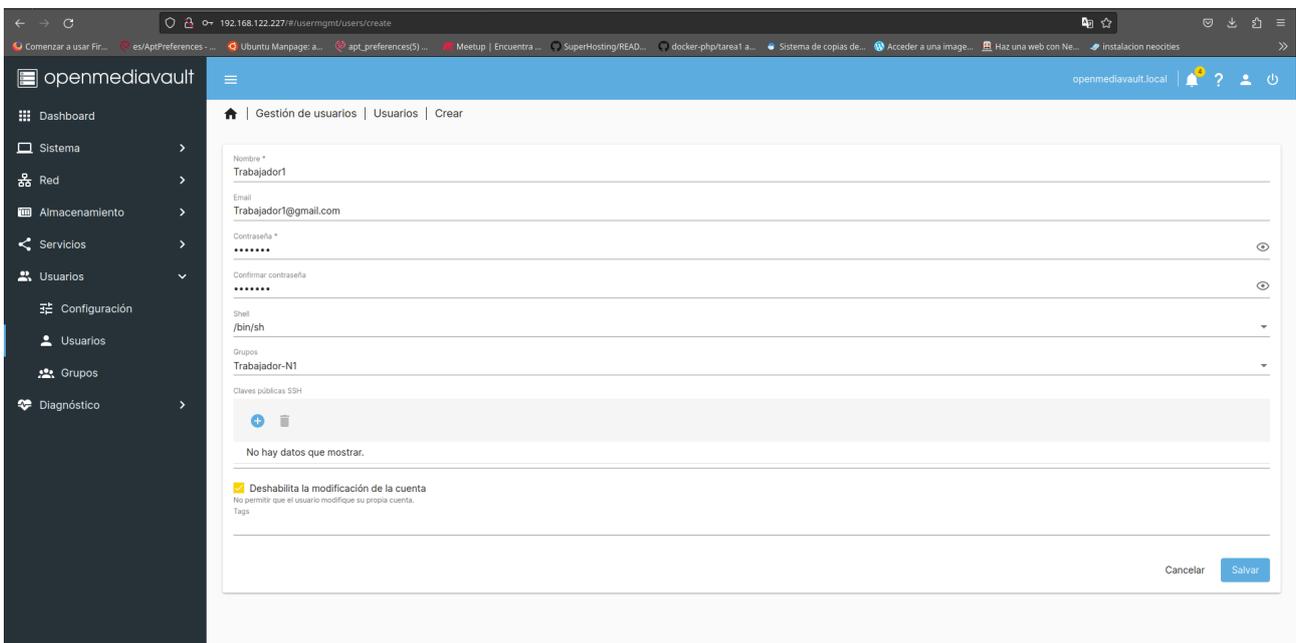
Primero, vamos a configurar los grupos de los usuarios que deseamos crear.



Creamos un primer grupo para los trabajadores del nivel 1 y otro grupo para el gerente.

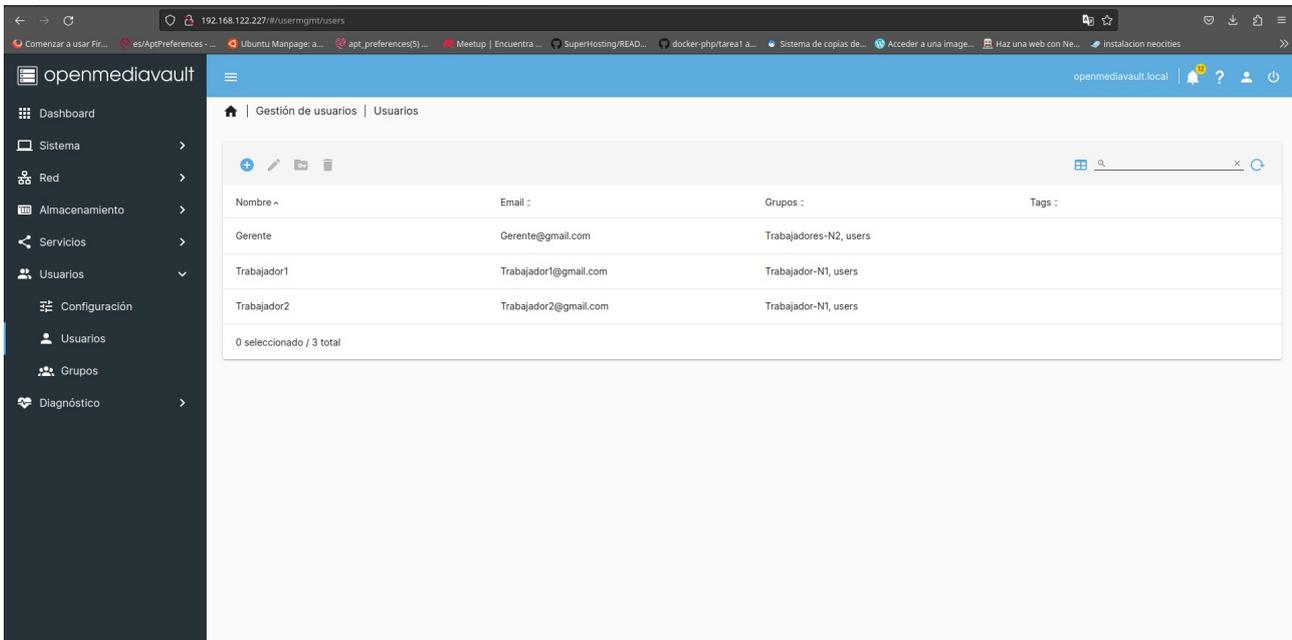


Una vez creado los grupos pasamos a crear los usuarios.



Hemos creado grupos porque nos proporcionan una mayor facilidad en la gestión de permisos de los usuarios al acceder a carpetas u otros recursos compartidos. Por ejemplo, si tenemos un conjunto de empleados que solo necesitan acceder a una carpeta compartida con ciertos privilegios, como permisos de solo lectura, podemos asignar esa carpeta al grupo al que pertenecen esos usuarios.

Esto simplifica la gestión en comparación con asignar permisos de manera individual a cada usuario.



4.3 Creación de RAID con BTRFS

Vamos a crear en nuestro servidor un RAID tipo espejo (RAID 1). Esta configuración nos proporcionará redundancia en los datos, lo que significa que tendremos dos discos duros de la misma capacidad. Esto es crucial, ya que para crear un RAID espejo los discos deben tener la misma capacidad. Con RAID 1, la misma información se almacenará en ambos discos duros, lo que garantiza una copia exacta de los datos en caso de fallo de uno de los discos.

¿Esto que significa?

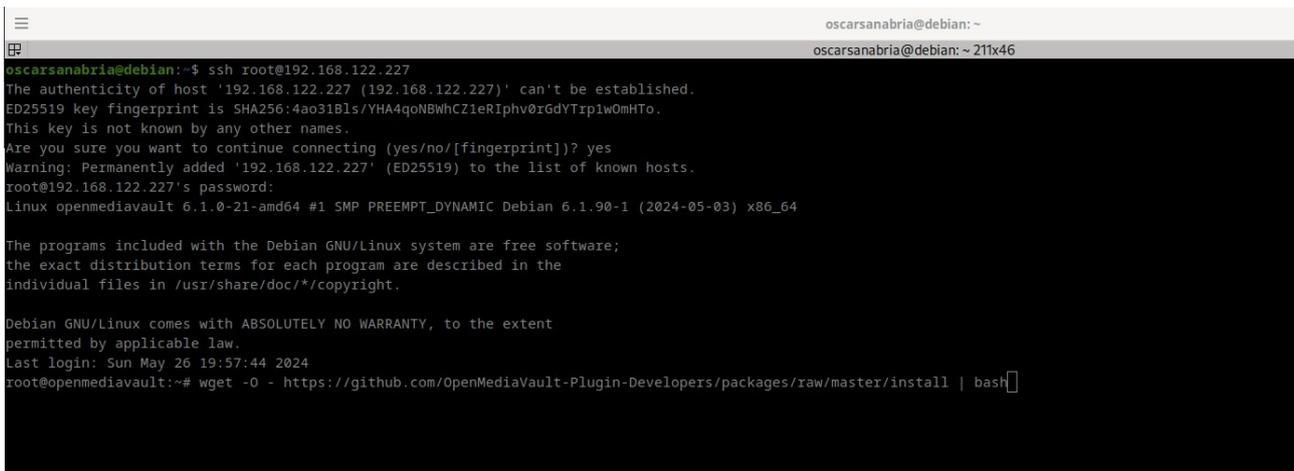
Al tener esta configuración realizada en nuestro servidor OMV, nos permite tener los datos más seguros, ya que si uno de los dos discos duros falla con el tiempo, nuestros datos seguirán estando disponibles en nuestro servidor gracias al otro disco duro que contiene la misma información. Además, esta configuración nos ofrece escalabilidad en el futuro, ya que si disponemos de más recursos económicos, podemos comprar más discos duros y ampliar este RAID sin problemas.

Si configuramos este RAID con el sistema de archivos Btrfs, obtenemos la capacidad de trabajar con grandes volúmenes de datos y redundancia de datos. Esta opción es una de las mejores en el mercado y actualmente es muy popular.

Ahora que hemos explicado brevemente esta configuración, procederemos a la creación de este RAID. Para ello, debemos instalar un plugin que nos permitirá realizar esta tarea. Usaremos el siguiente comando para instalar el apartado de plugins en nuestro servidor, lo cual nos dará acceso a

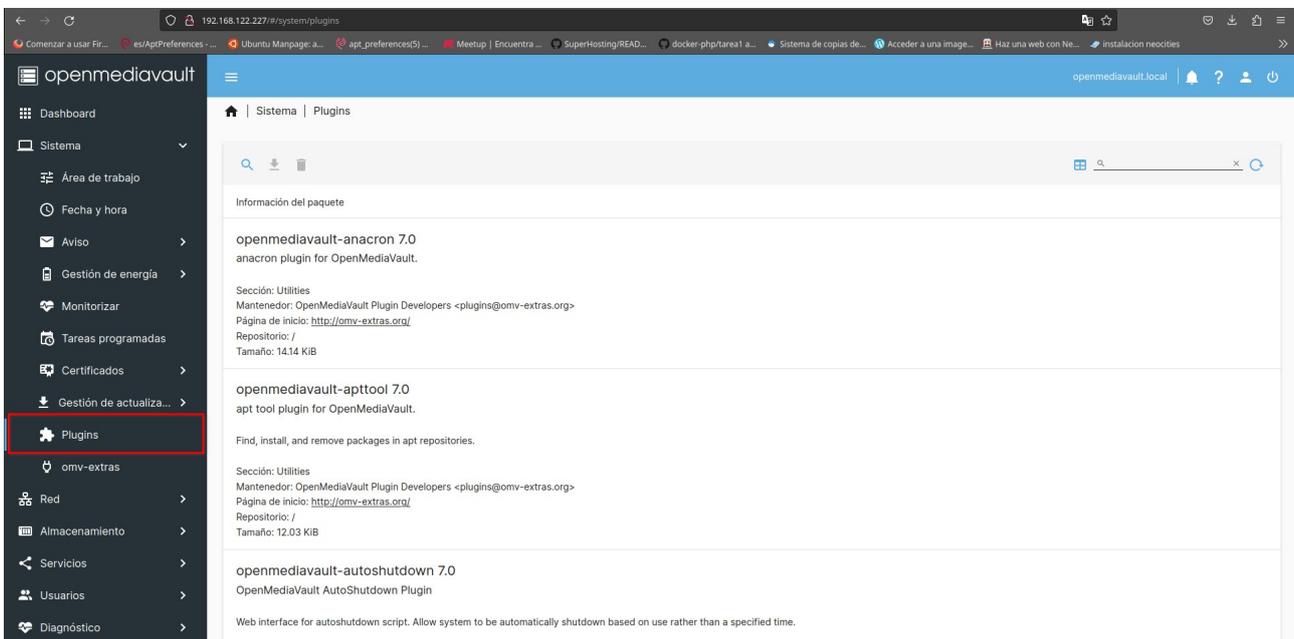
estas maravillosas funcionalidades que ofrece OMV. Nos conectaremos al servidor mediante una conexión SSH como usuario root para poder ejecutar este comando.

```
wget -O -  
https://github.com/OpenMediaVault-Plugin-Developers/packages/raw/master/install | bash
```

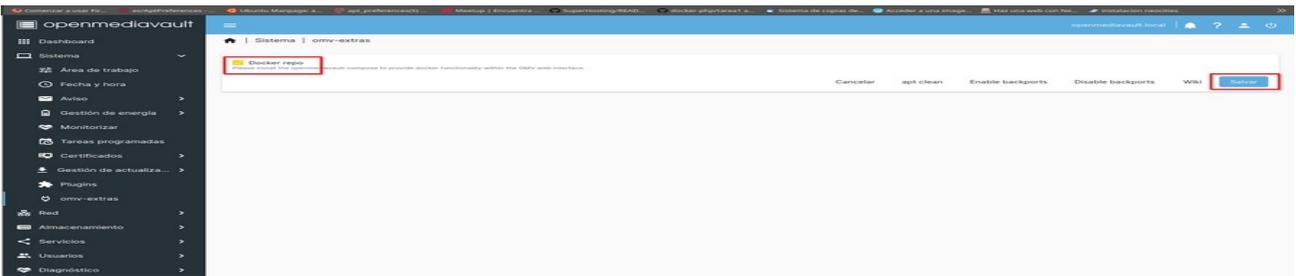


```
oscarsanabria@debian:~  
oscarsanabria@debian: ~ 21x46  
oscarsanabria@debian:~$ ssh root@192.168.122.227  
The authenticity of host '192.168.122.227 (192.168.122.227)' can't be established.  
ED25519 key fingerprint is SHA256:4ao318ls/YHA4qoNBWhCZ1eRlphv0rGdYTrp1w0mHTo.  
This key is not known by any other names.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '192.168.122.227' (ED25519) to the list of known hosts.  
root@192.168.122.227's password:  
Linux openmediavault 6.1.0-21-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.90-1 (2024-05-03) x86_64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Sun May 26 19:57:44 2024  
root@openmediavault:~# wget -O - https://github.com/OpenMediaVault-Plugin-Developers/packages/raw/master/install | bash
```

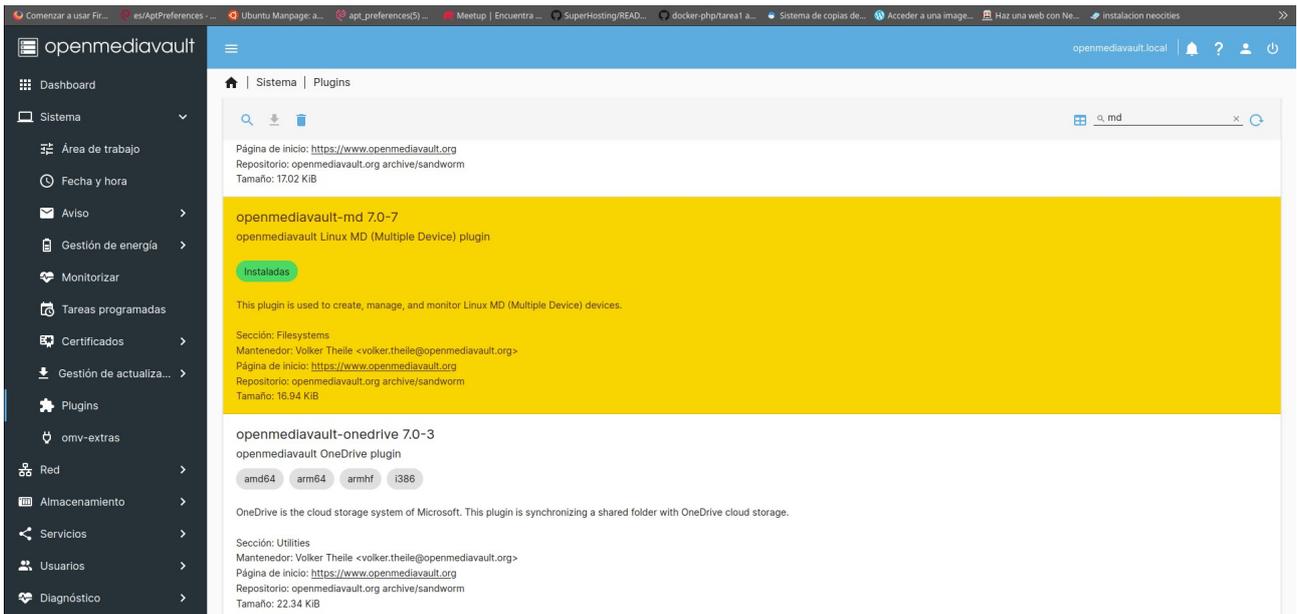
Una vez finalizado este comando nos dirigimos al panel del servidor y se nos debera de habilitar una opción como esta.



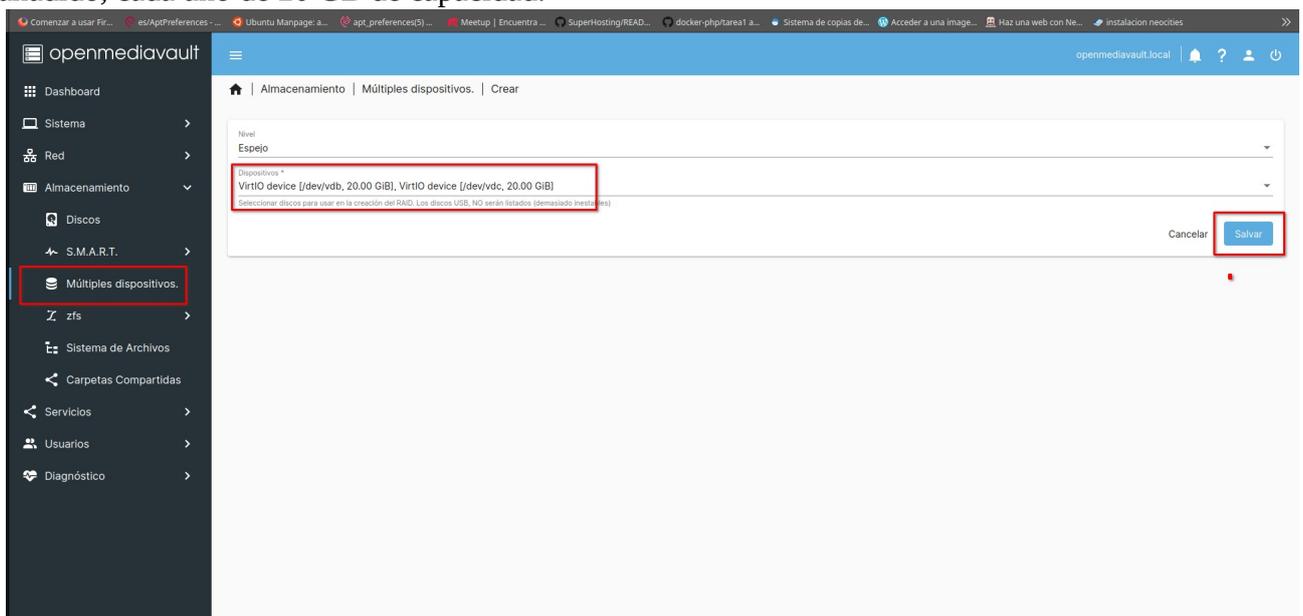
Pero aquí no termina la configuración, no debemos de ir a la opción omv-extras y realizar lo siguiente.

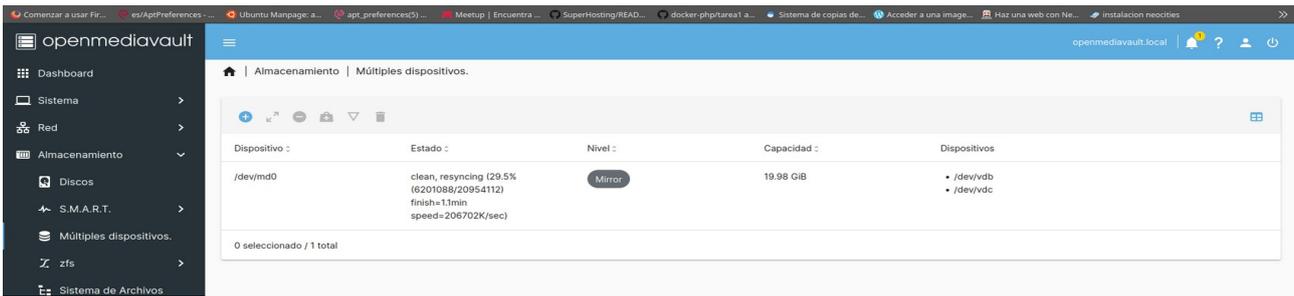


Una vez completados los pasos anteriores, procedemos a instalar el plugin que nos permitirá la creación de un RAID en espejo.

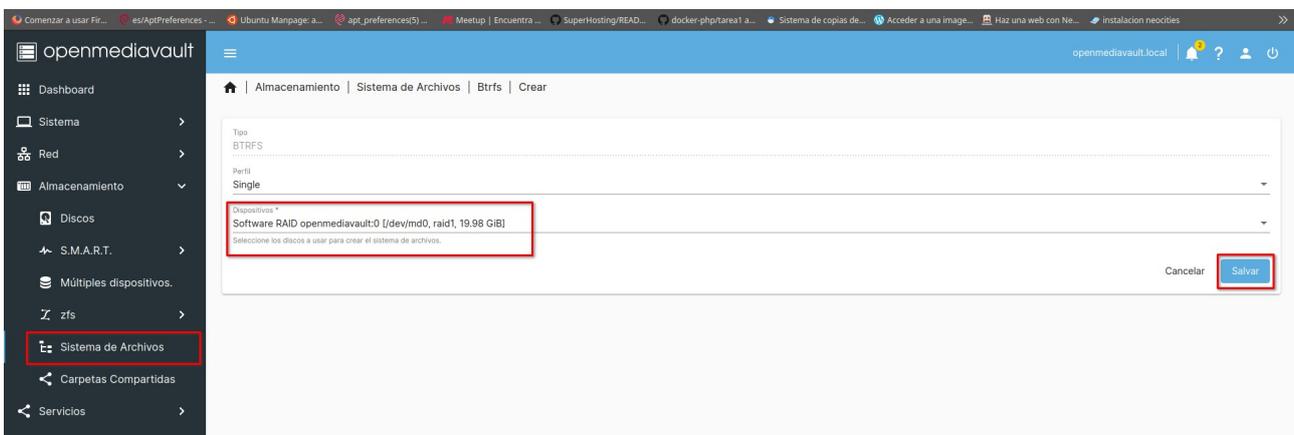


Una vez instalado el plugin, procedemos a crear el RAID y seleccionamos los dos discos que hemos añadido, cada uno de 20 GB de capacidad.

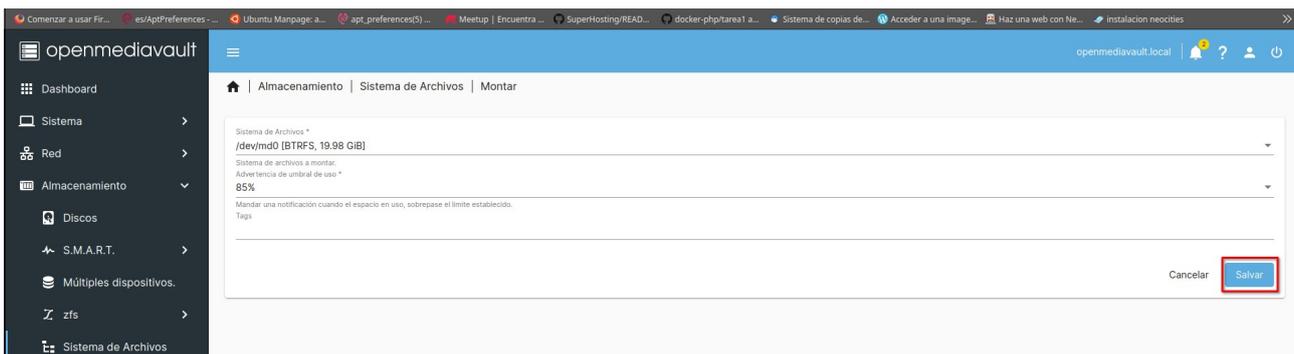




A continuación, formateamos el RAID con el sistema de archivos indicado anteriormente y lo montamos.



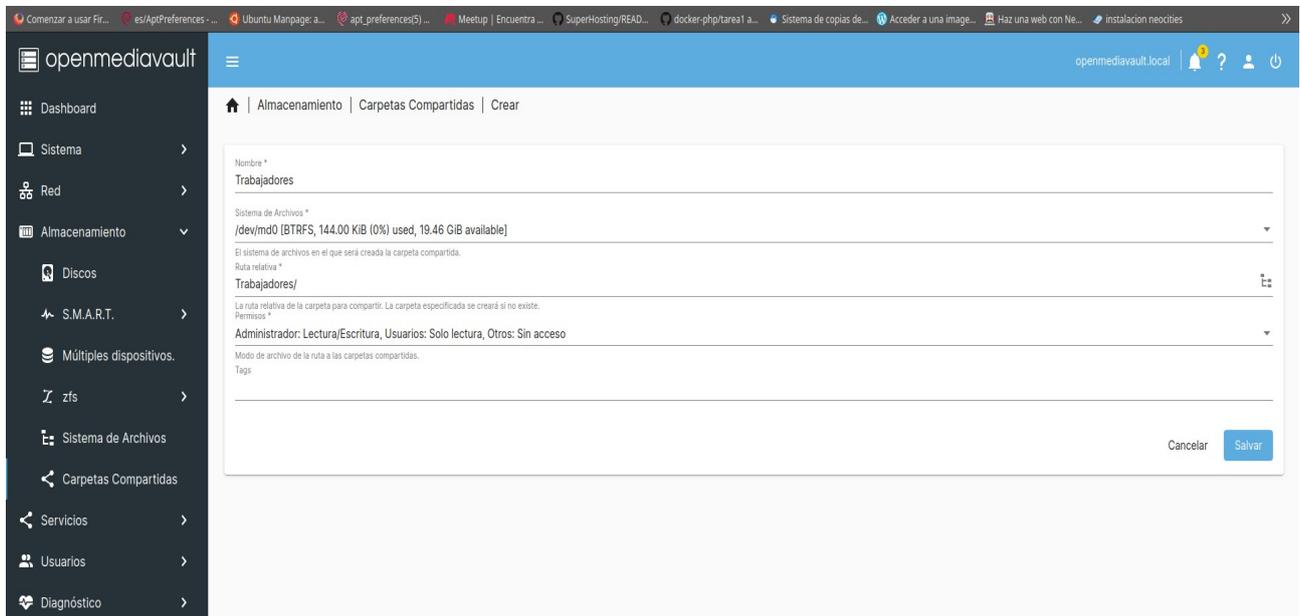
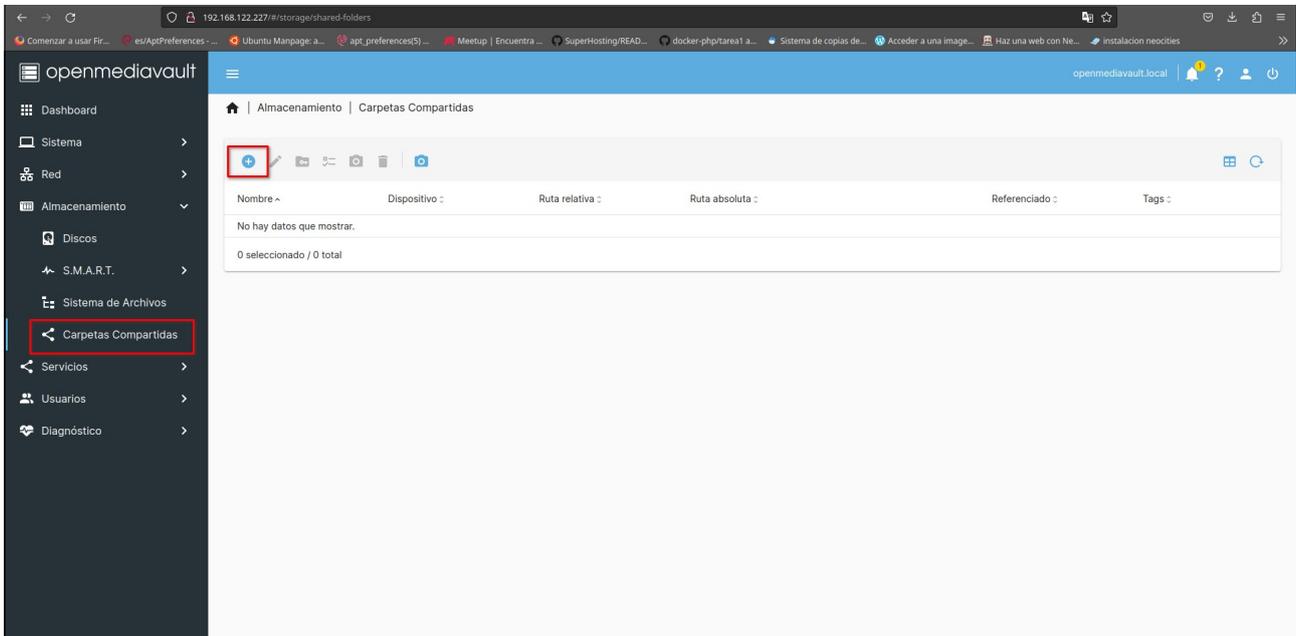
A cierta capacidad, el propio servidor nos notificará cuando el disco duro esté lleno, permitiéndonos tener constancia de cuánta capacidad estamos utilizando. Esto es importante para evitar problemas de falta de espacio en disco y garantizar el funcionamiento adecuado del servidor.



4.4 Creación de carpeta y compartir en la red

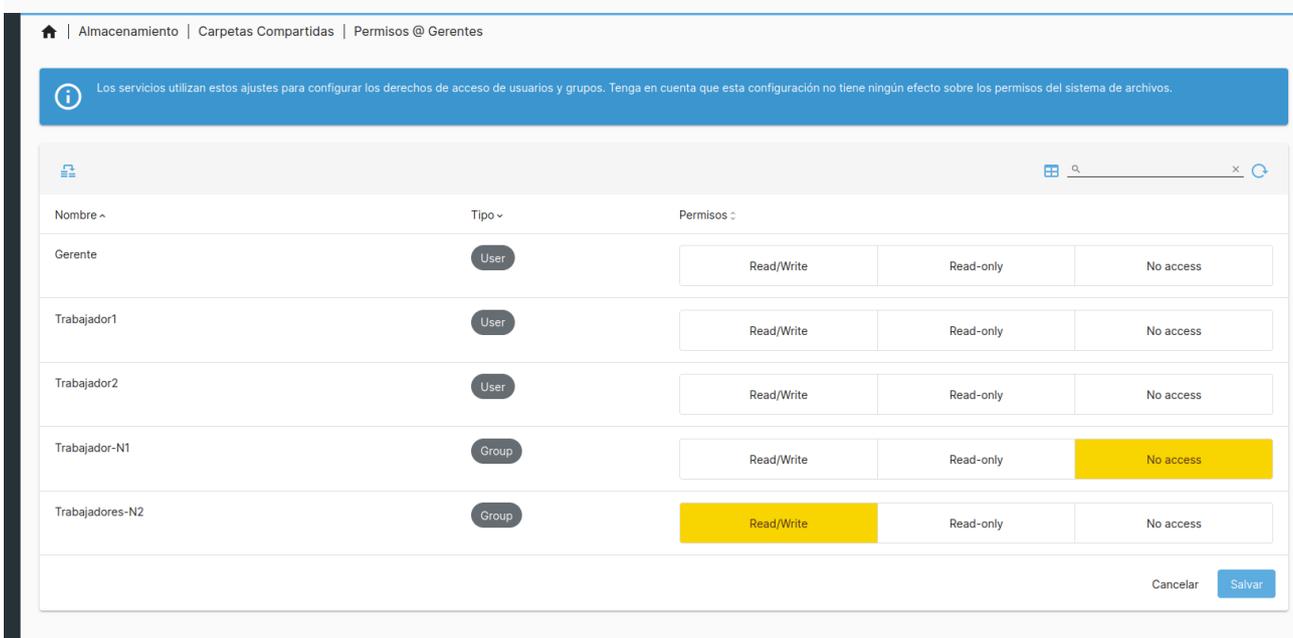
Para compartir nuestras carpetas creadas en el servidor, necesitaremos el siguiente servicio: SMB. Con este servicio, podremos compartir carpetas tanto con clientes Linux como con Windows.

Veamos los pasos que debemos seguir para llevar esto a cabo. Lo primero que debemos hacer es crear dos carpetas. En mi caso, crearé dos ejemplos para ilustrar el proceso.

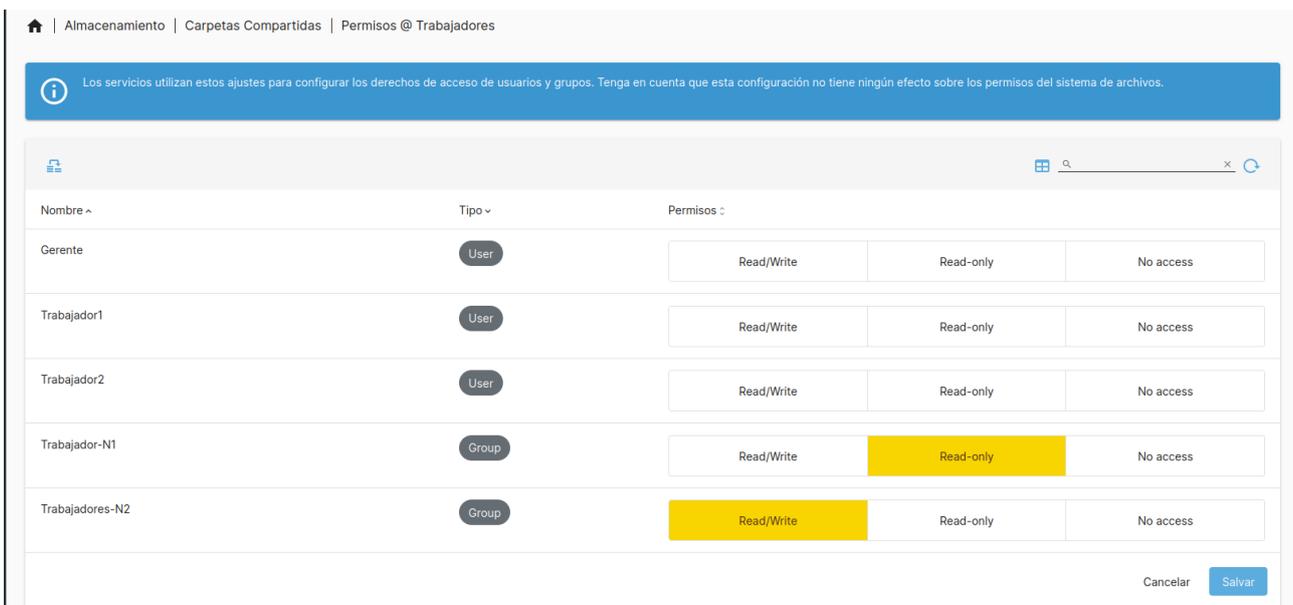


Si recordáis, anteriormente cuando creamos los usuarios, hablamos sobre los permisos y grupos. Aquí es donde llevaremos a cabo estas acciones sobre las carpetas.

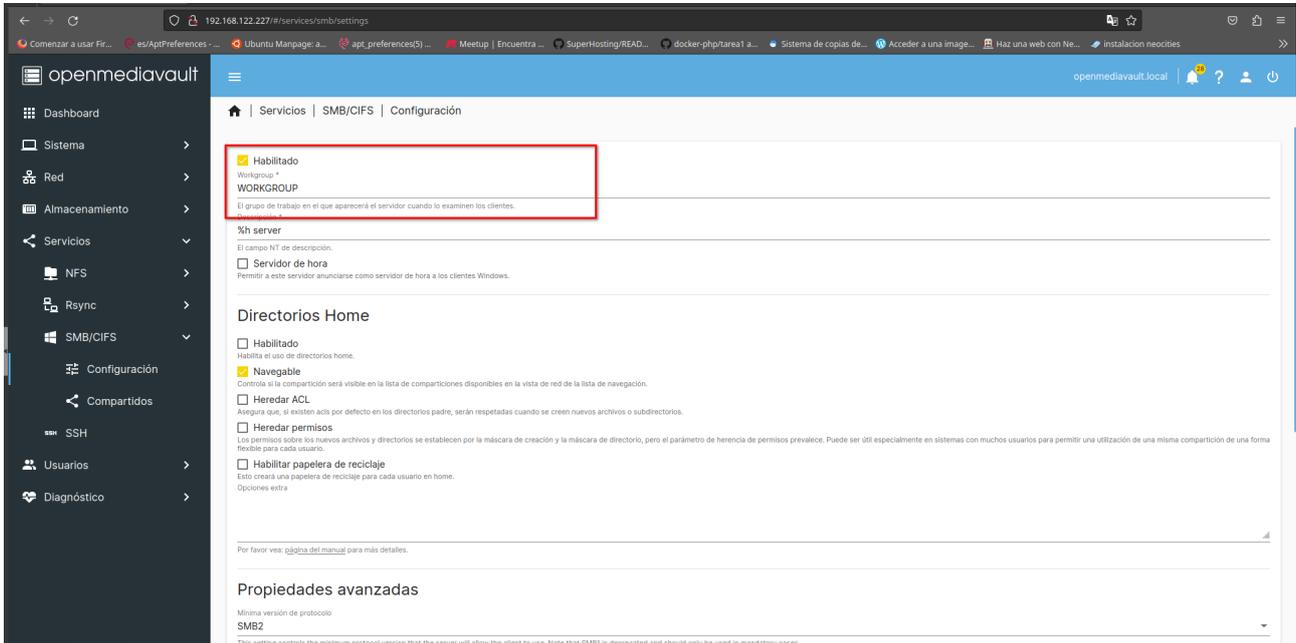
El gerente tendrá permisos para modificar tanto su propia carpeta como la de los trabajadores. Sin embargo, los trabajadores no tendrán acceso a la carpeta del gerente.



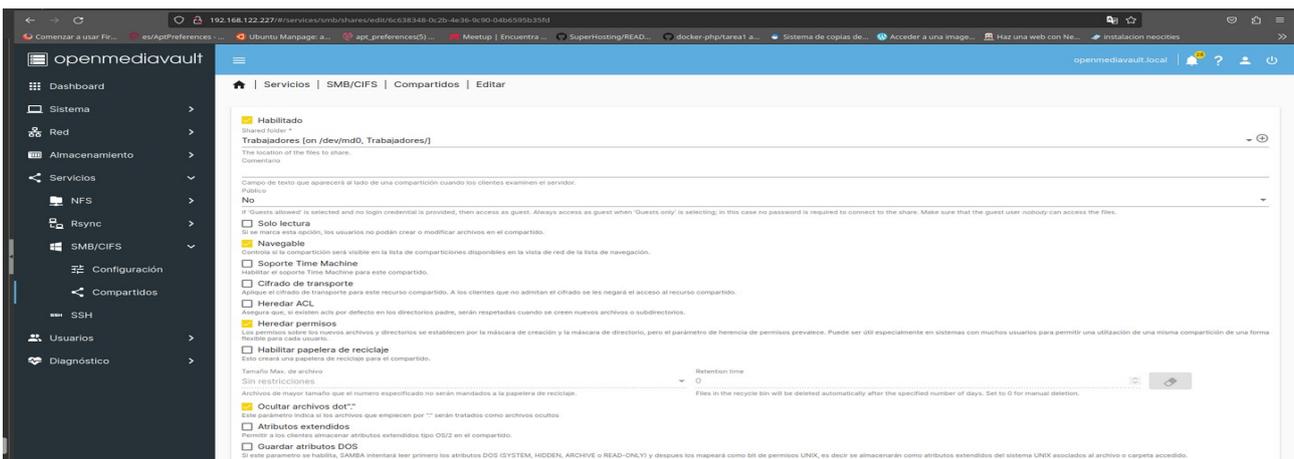
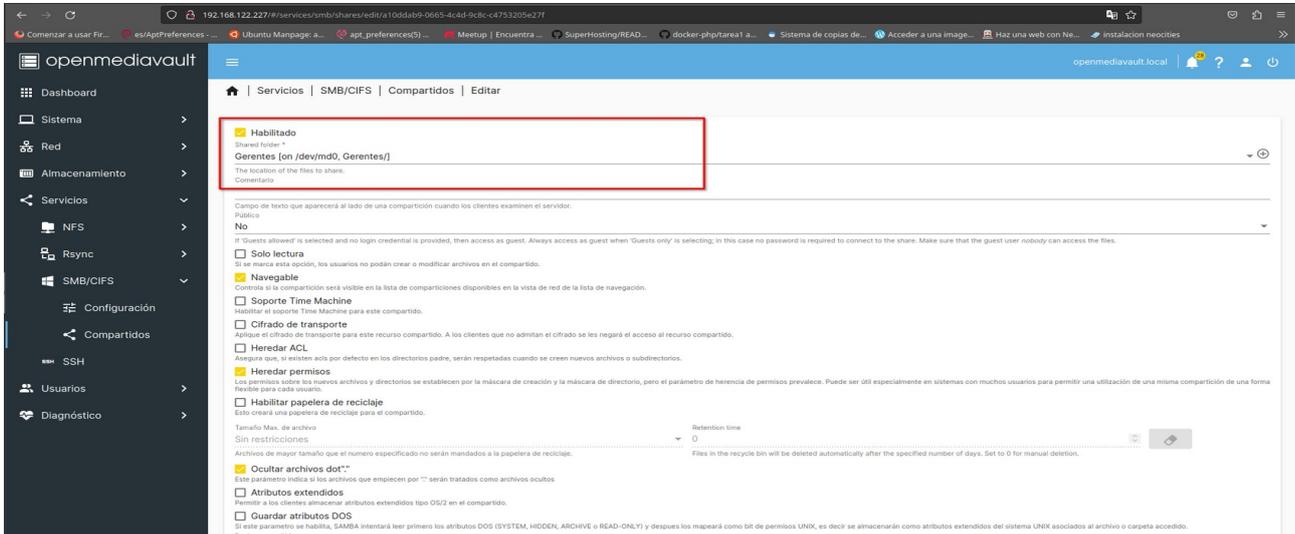
Los trabajadores solo podrán ver su propia carpeta, sin poder modificar su contenido. Por otro lado, el gerente tendrá permisos para modificar tanto su propia carpeta como las de los trabajadores.



Una vez creadas estas carpetas, vamos a compartirlas utilizando el servicio Samba, ya que es muy común y conveniente. Sin embargo, primero debemos habilitar el servicio. Si no lo hacemos, cuando compartamos las carpetas, es posible que no funcione correctamente.

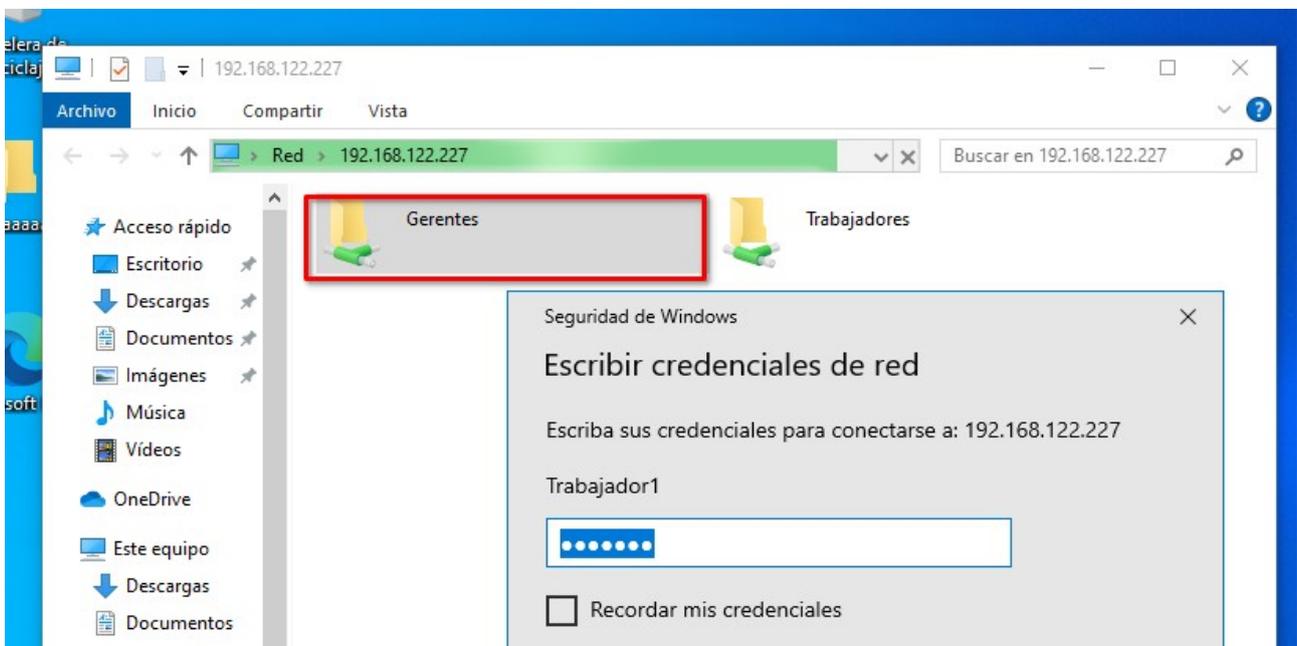
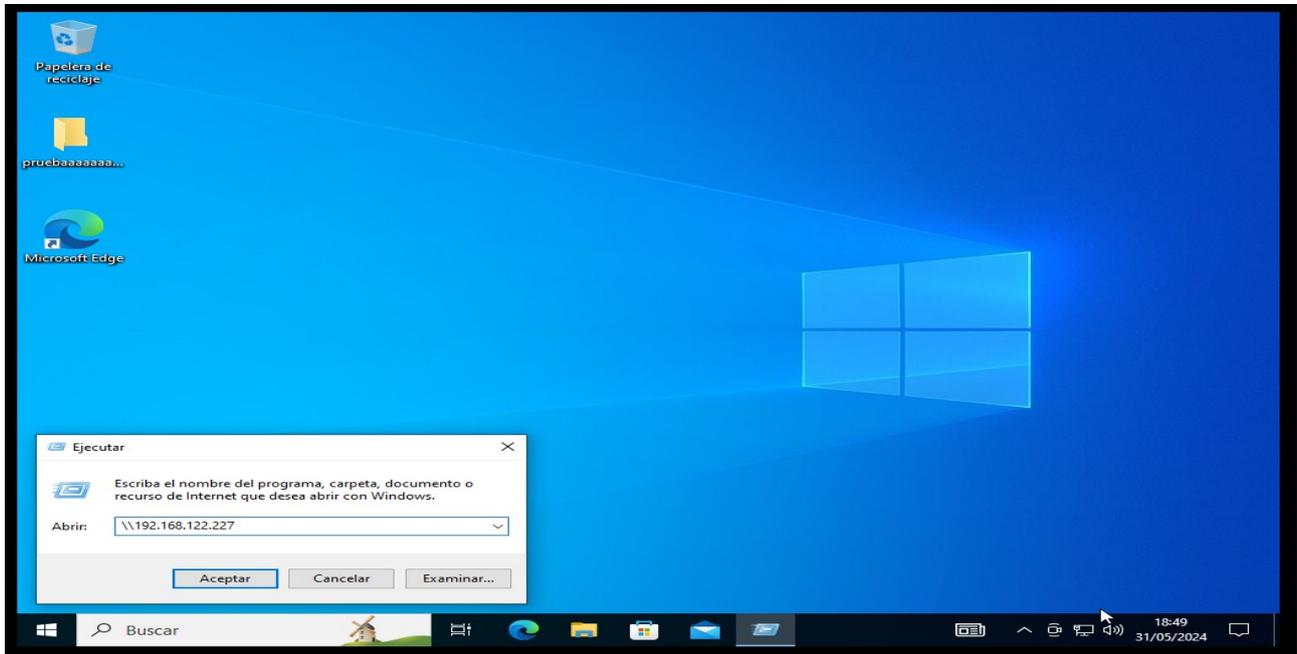


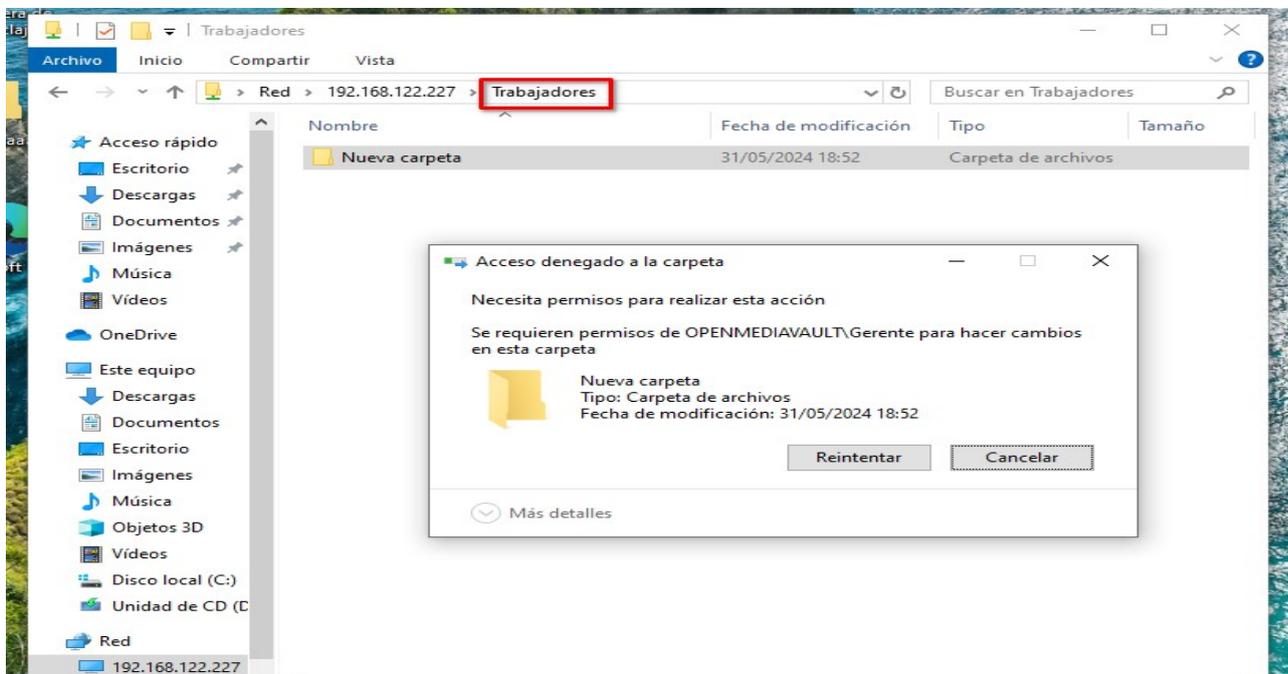
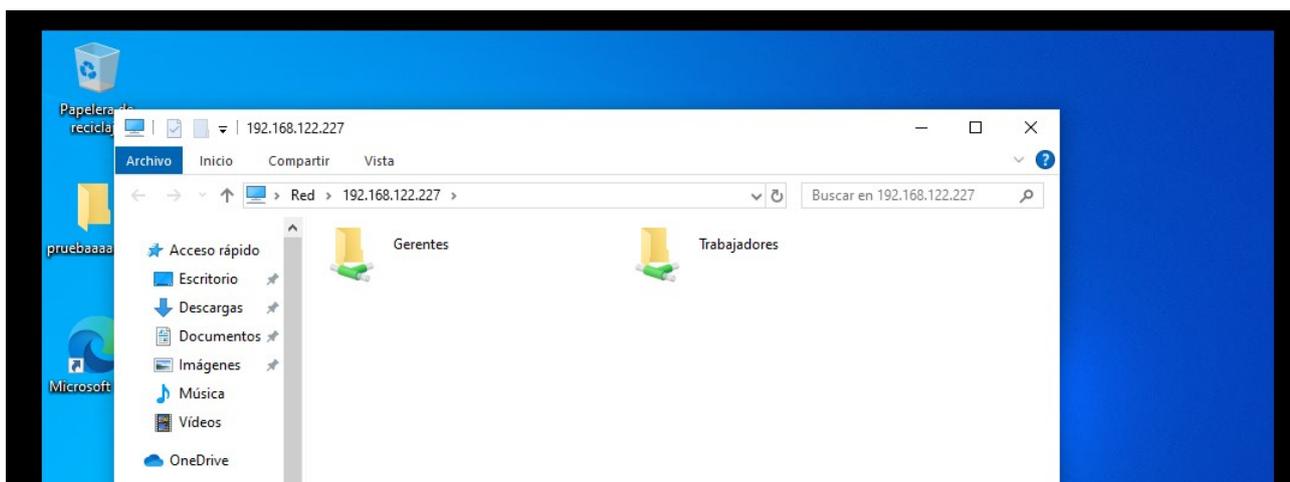
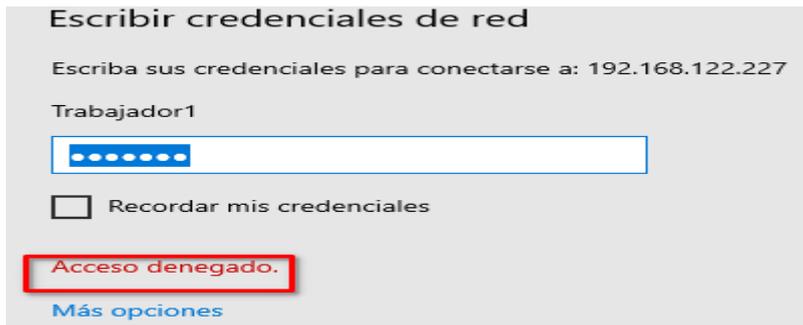
Una vez habilitado guardamos y configuramos las carpetas.



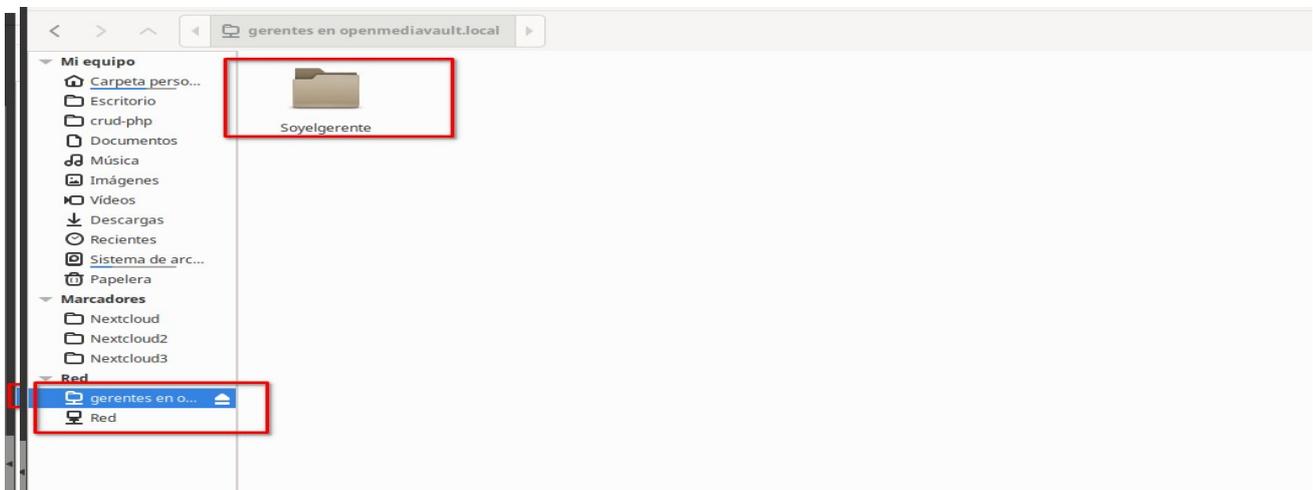
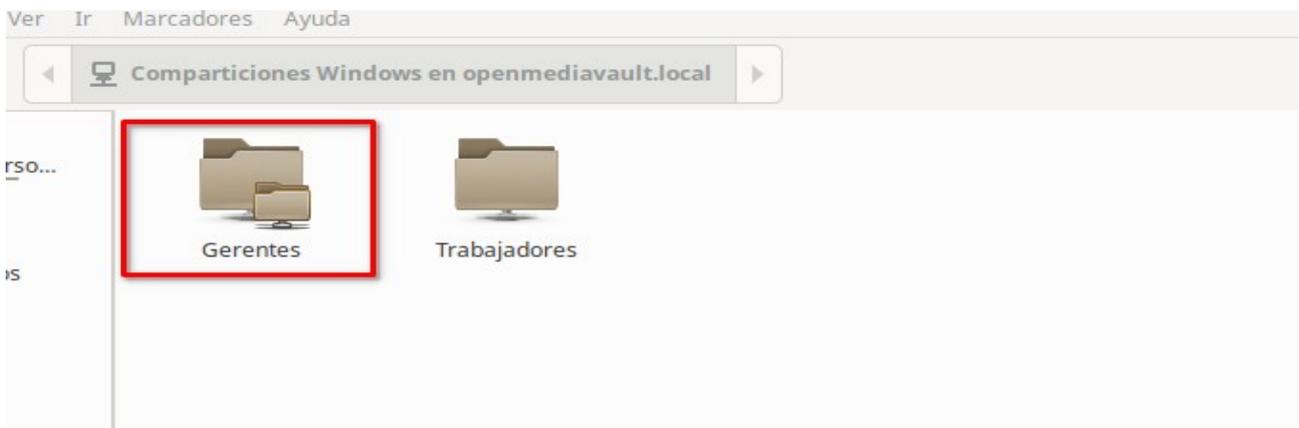
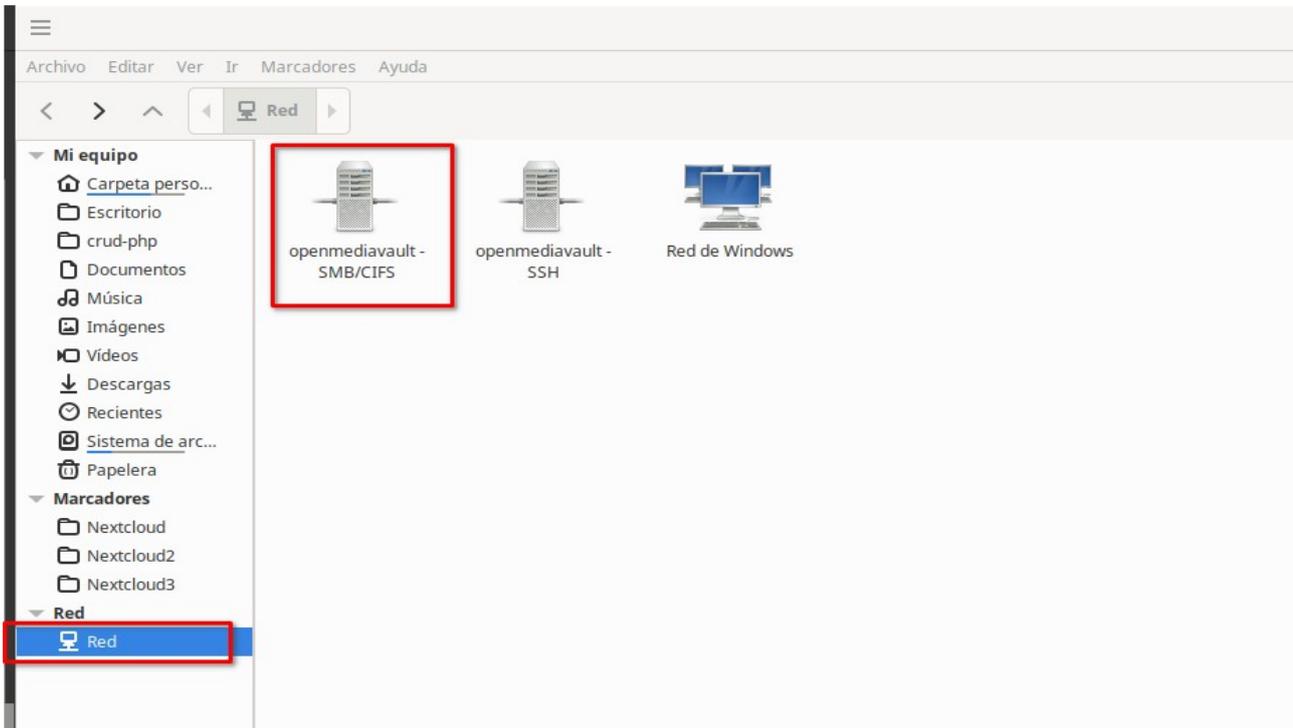
4.4.1 Pruebas

- Cliente Windows usuario Trabajador1:





- Cliente Linux usuario Gerente:

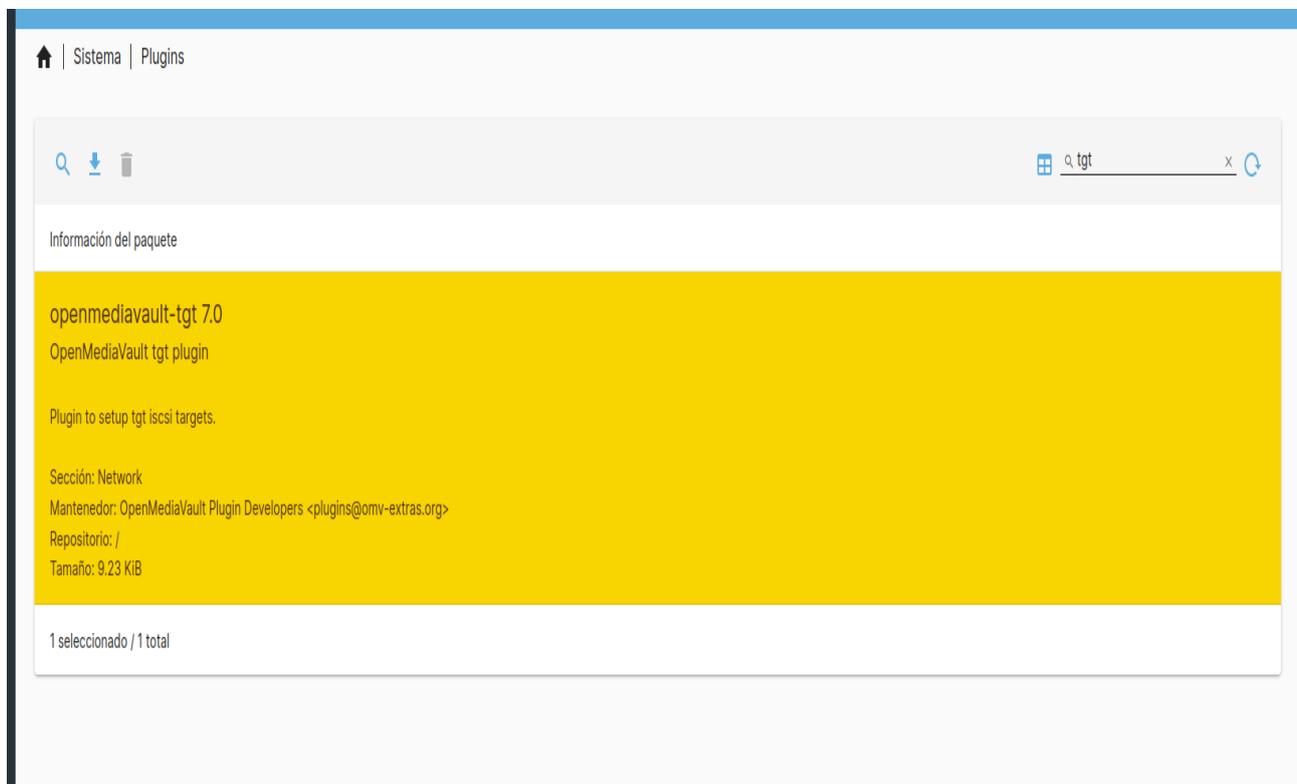


4.5 Configuración de iSCSI con LVM- Cliente Linux.

El protocolo iSCSI permite acceder a dispositivos de bloques a través de redes TCP/IP, compartiendo discos duros completos que los clientes pueden gestionar. Es más económico que Fibre Channel y compatible con la mayoría de los sistemas operativos, permitiendo conectar un dispositivo a varios clientes simultáneamente. En este contexto, configuraremos iSCSI en nuestro servidor OMV para compartir discos duros en la red. Esta configuración es robusta, eficiente, flexible, y fácil de gestionar, lo que la convierte en una excelente opción para almacenamiento en red.

- **Unidad lógica (LUN):** Dispositivo de bloques compartido por el servidor iSCSI (discos duros, particiones, volúmenes lógicos).
- **Target:** Recurso compartido desde el servidor que incluye una o varias LUNs. Puede agrupar varios discos duros y permitir que un cliente acceda a ellos mediante una única conexión.
- **Initiator:** Cliente iSCSI que se conecta al servidor.
- **Multipath:** Asegura la disponibilidad del dispositivo remoto mediante múltiples rutas entre el target y el initiator, manteniendo la conexión si una ruta falla.
- **IQN:** Formato para describir recursos compartidos, usando el esquema iqn.(año)-(mes).(dominio invertido):(nombre único), por ejemplo, iqn.2021-02.com.ejemplo

Lo primero que debemos hacer es instalar los siguientes plugins que nos permitirán llevar a cabo este tipo de configuraciones.



🏠 | Sistema | Plugins

🔍 ⬇️ 🗑️ 🏠 tgt x ↻

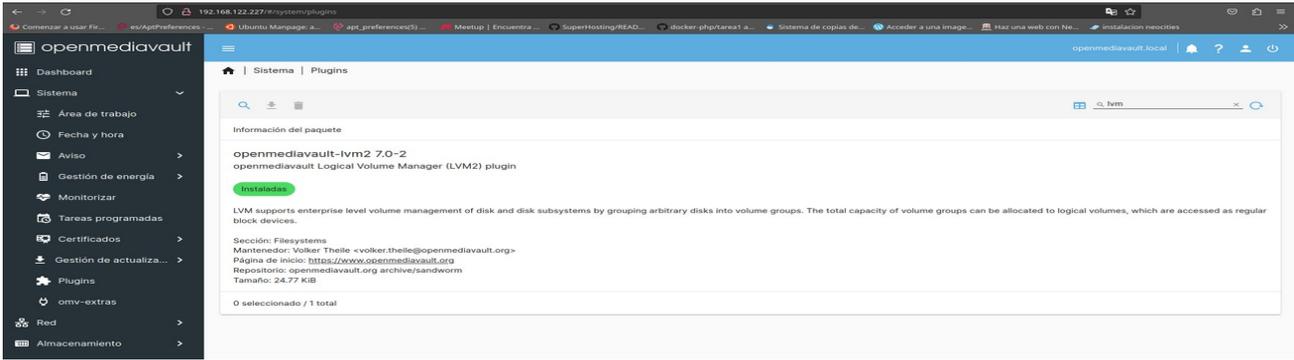
Información del paquete

openmediavault-tgt 7.0
OpenMediaVault tgt plugin

Plugin to setup tgt iscsi targets.

Sección: Network
Mantenedor: OpenMediaVault Plugin Developers <plugins@omv-extras.org>
Repositorio: /
Tamaño: 9.23 KiB

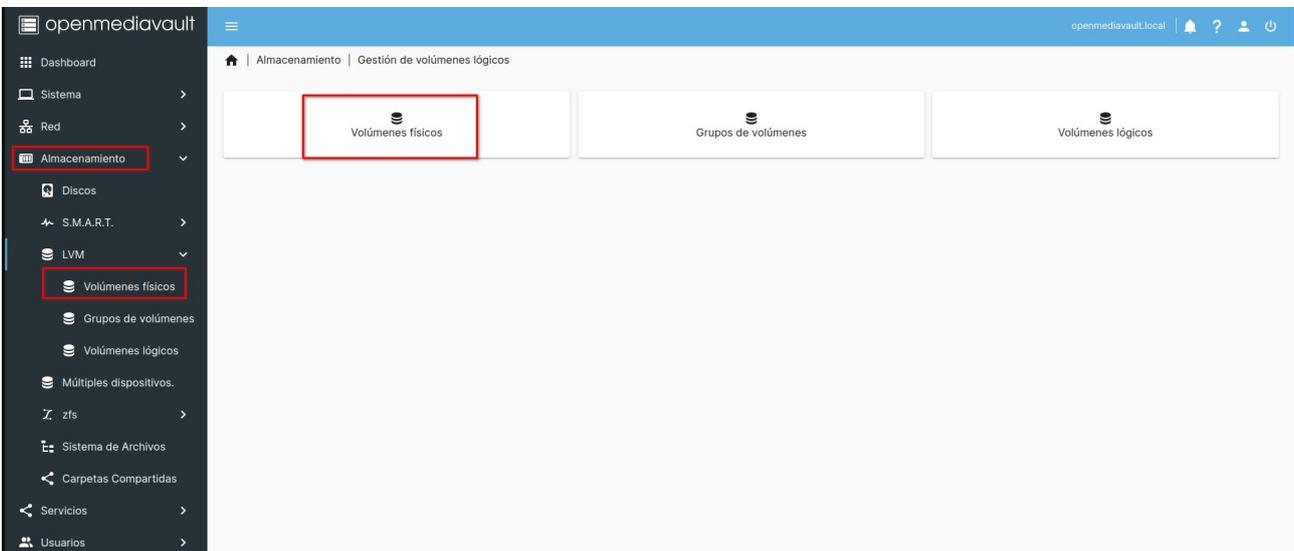
1 seleccionado / 1 total

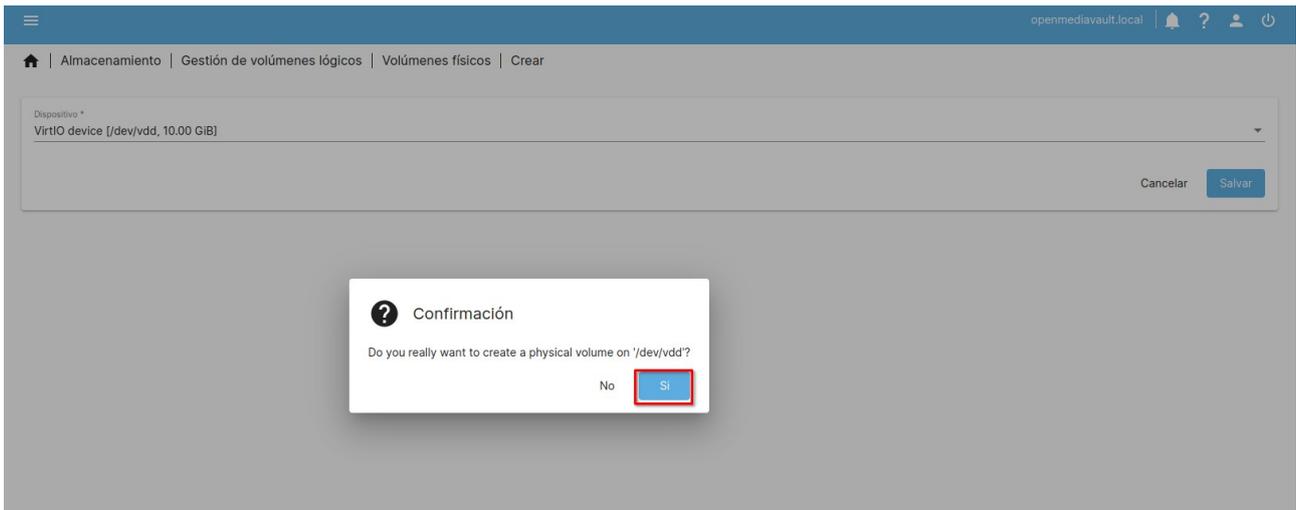
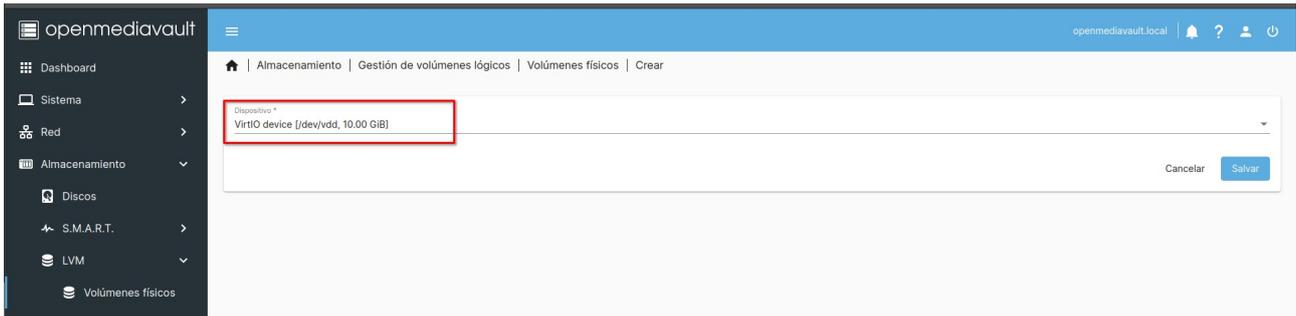


Una vez instalados, encontraremos estos plugins en la sección de servicios y almacenamiento. Para ello, vamos a configurar primero el LVM en nuestro servidor OMV. Añadiré un disco de 10 GB, que compartiremos con el cliente Linux.

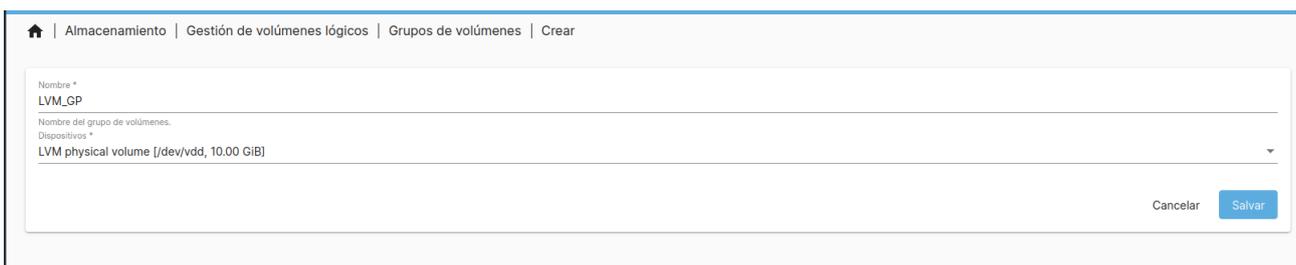
The screenshot shows the 'Gestión de volúmenes lógicos' page in the OpenMediaVault interface. It features a table with the following columns: 'Dispositivo', 'Modelo', 'Número de Serie', 'Vendedor', and 'Capacidad'. There are four rows of data, with the last row highlighted by a red border. Below the table, it indicates '0 seleccionado / 4 total'.

Dispositivo	Modelo	Número de Serie	Vendedor	Capacidad
/dev/vda			0x1af4	20.00 GiB
/dev/vdb			0x1af4	20.00 GiB
/dev/vdc			0x1af4	20.00 GiB
/dev/vdd			0x1af4	10.00 GiB





Tras seleccionar el disco físico donde deseamos realizar nuestro LVM, pasamos a crear el grupo de volúmenes. Es importante guardar el nombre del grupo en un bloc de notas, ya que lo necesitaremos para la configuración iSCSI. Nombre del grupo: LVM_GP.



Nombre	Disponible	Libre	Volúmenes físicos	Volúmenes lógicos
LVM_GP	10.00 GiB	10.00 GiB	/dev/vdd	

0 seleccionado / 1 total

Y por último, creamos los volúmenes lógicos. Como mencioné anteriormente, también debemos guardar el nombre del volumen lógico. Nombre del volumen lógico: LOGICA_VOLUMEN.

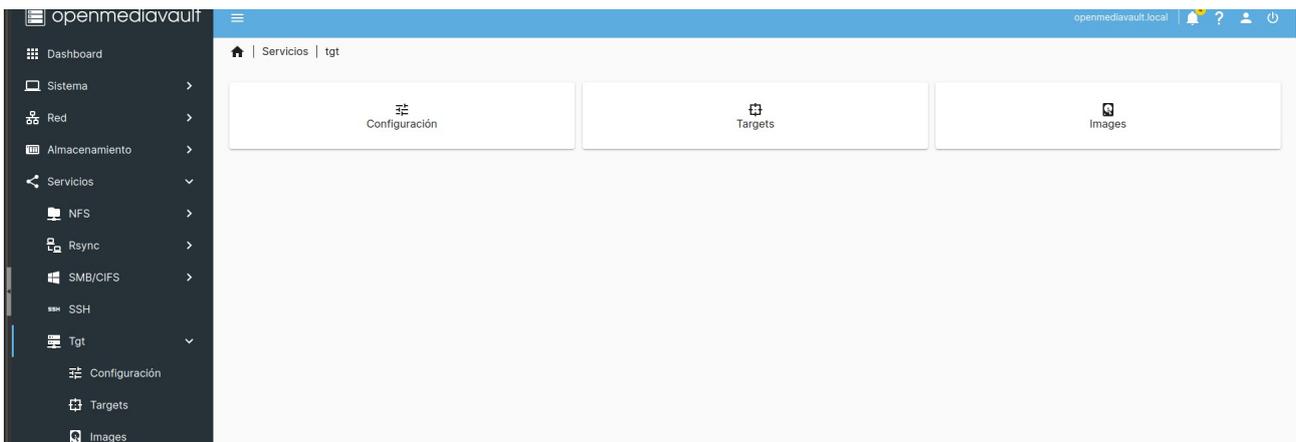
Nombre *
LOGICA_VOLUMEN
Nombre del volumen lógico.
Grupo de volúmenes *
LVM volume group LVM_GP [/dev/LVM_GP, 9.99 GiB, 9.99 GiB free]
Size
The percentage of the total space in the volume group to allocate for the new logical volume.
Cancelar Salvar

En mi caso, le he asignado la capacidad máxima de este disco, pero podemos ajustarla usando la barra que encontramos en el último paso.

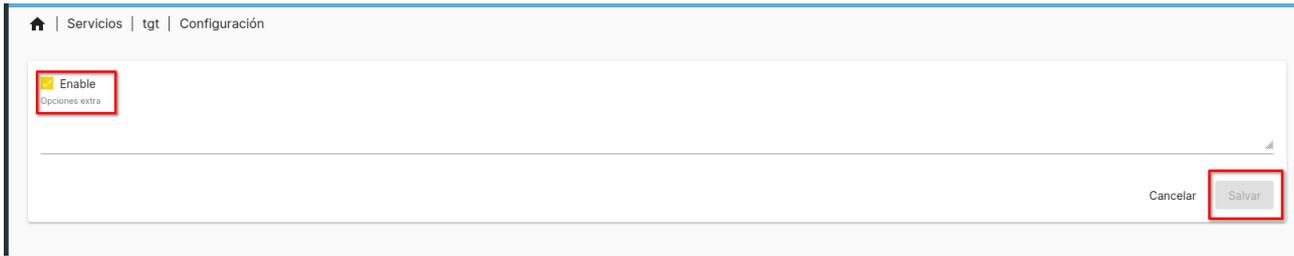
Nombre	Capacidad	Grupo de volúmenes	Activo
LOGICA_VOLUMEN	10.00 GiB	LVM_GP	✓

0 seleccionado / 1 total

Una vez finalizado, debería quedarnos así: LVM_GP-LOGICA_VOLUMEN. Esto lo utilizaremos para la siguiente configuración. Ahora, nos dirigimos a la configuración de iSCSI.

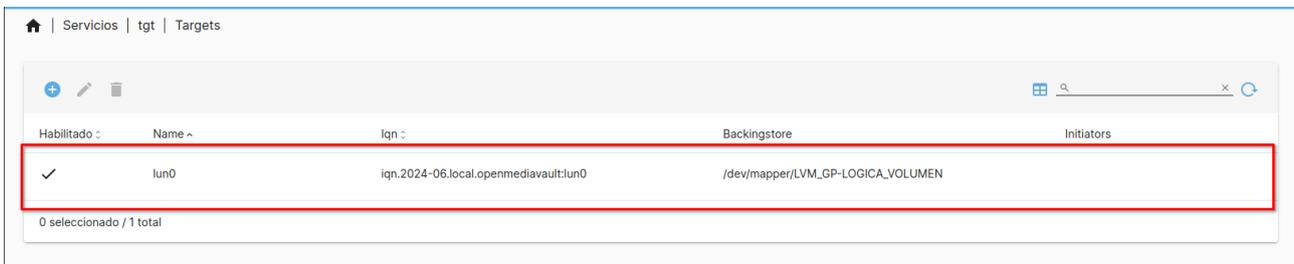


Activamos el servicios y guardamos.



Empezamos añadir un target.

```
root@openmediavault:~# ls /dev/mapper/  
control LVM_GP-LOGICA_VOLUMEN  
root@openmediavault:~# _
```



Antes de añadir un target para el cliente Linux, debemos realizar algunas configuraciones en el propio cliente. Primero, instalaremos el iniciador de iSCSI, que nos permitirá utilizar el target del servidor.

- **sudo apt-get install open-iscsi**

```
oscarosanabria@debian: ~  
oscarosanabria@debian:~$ sudo apt-get install open-iscsi  
[sudo] contraseña para oscarosanabria:  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias... Hecho  
Leyendo la información de estado... Hecho  
Se instalarán los siguientes paquetes adicionales:  
libisns0 libopeniscsiusr  
Paquetes recomendados:  
finalrd  
Se instalarán los siguientes paquetes NUEVOS:  
libisns0 libopeniscsiusr open-iscsi  
0 actualizados, 3 nuevos se instalarán, 0 para eliminar y 217 no actualizados.  
Se necesita descargar 464 kB de archivos.  
Se utilizarán 2.152 kB de espacio de disco adicional después de esta operación.  
¿Desea continuar? [S/n]
```

Editamos el fichero `/etc/iscsi/iscsid.conf` para que busque los target automáticamente.

- **sudo nano /etc/iscsi/iscsid.conf**

```
ooscarsanabria@debian: ~
ooscarsanabria@debian: ~ 21x46
/etc/iscsi/iscsid.conf *
GNU nano 7.2
#
# Open-iSCSI default configuration.
# Could be located at /etc/iscsi/iscsid.conf or ~/.iscsid.conf
#
# Note: To set any of these values for a specific node/session run
# the iscsiadm --mode node --op command for the value. See the README
# and man page for iscsiadm for details on the --op command.
#
#####
# iscsid daemon config
#####
#
# If you want iscsid to start the first time an iscsi tool
# needs to access it, instead of starting it when the init
# scripts run, set the iscsid startup command here. This
# should normally only need to be done by distro package
# maintainers. If you leave the iscsid daemon running all
# the time then leave this attribute commented out.
#
# Default for Fedora and RHEL. Uncomment to activate.
# iscsid.startup = /bin/systemctl start iscsid.socket iscsiui.socket
#
# Default for Debian and Ubuntu. Uncomment to activate.
iscsid.startup = automatic
#
# Default if you are not using systemd. Uncomment to activate.
# iscsid.startup = /usr/bin/service start iscsid
#
# Check for active mounts on devices reachable through a session
# and refuse to logout if there are any. Defaults to "No".
# iscsid.safe_logout = Yes
```

Guardamos la configuración y reiniciamos nuestro servicio.

- **sudo systemctl restart open-iscsi**

```
ooscarsanabria@debian: ~$ sudo systemctl restart open-iscsi
ooscarsanabria@debian: ~$
```

Con el siguiente comando vamos a escanear para buscar el target del servidor, el cual le debemos de indicar la ip del servidor en el siguiente comando.

- **sudo iscsiadm -m discovery -t st -p 192.168.122.227**

```
ooscarsanabria@debian: ~$ sudo iscsiadm -m discovery -t st -p 192.168.122.227
192.168.122.227:3260,1 iqn.2024-06.local.openmediavault:lun0
ooscarsanabria@debian: ~$
```

Para conectarnos a este target que se ha detectado en el servidor, haremos lo siguiente. Utilizaremos iSCSI, que es la herramienta que contiene el paquete instalado anteriormente para su administración. Tenemos que pasarle varios parámetros, como el modo, nombre del target y el portal del destino, que sería la IP del servidor. El puerto por defecto es el 3260.

- **sudo iscsiadm -m node -T iqn.2024-06.local.openmediavault:lun1 --portal "192.168.122.227" --login**

```

oscarsanabria@debian: ~
oscarsanabria@debian: ~ 211x46
oscarsanabria@debian:~$ sudo iscsiadm -m node -T iqn.2024-06.local.openmediavault:lun1 --portal "192.168.122.227" --login
Logging in to [iface: default, target: iqn.2024-06.local.openmediavault:lun0, portal: 192.168.122.227,3260]
Login to [iface: default, target: iqn.2024-06.local.openmediavault:lun0, portal: 192.168.122.227,3260] successful.
oscarsanabria@debian:~$

```

Podemos comprobar que se a conectado con éxito. Si todo ha ido bien deberíamos de ver el nuevo disco.

```

oscarsanabria@debian: ~
oscarsanabria@debian: ~ 211x46
oscarsanabria@debian:~$ lsblk -f
NAME        FSTYPE     FSVER     LABEL UUID                                FSAVAIL FSUSE% MOUNTPOINTS
loop0
loop1
loop2
loop3
loop4
loop5
loop6
sda
nvme0n1
├─nvme0n1p1  vfat       FAT32     E4D7-CBA6                                480,2M   1% /boot/efi
├─nvme0n1p2  LVM2_member LVM2 001   cmiyxt-Z0Dd-lwc0-EK5K-pGY0-iyzE-pt166a
│   └─vg--01-Raiz ext4       1.0      3892e1e0-53a0-4b31-b541-9592b1a1af99   125,9G   26% /
│   └─vg--01-Var  ext4       1.0      20f8045f-58f6-4460-ad9a-fc3ecc2ccf26   44,7G    54% /var
│   └─vg--01-Home ext4       1.0      05187b6e-cb01-41cb-9116-164ae394afc8   49,9G    66% /home

```

A continuación procedemos al formatear y montar el disco nuevo.

- **sudo fdisk /dev/sda**

```

oscarsanabria@debian:~$ sudo fdisk /dev/sda
Bienvenido a fdisk (util-linux 2.38.1).
Los cambios solo permanecerán en la memoria, hasta que decida escribirlos.
Tenga cuidado antes de utilizar la orden de escritura.

El dispositivo no contiene una tabla de particiones reconocida.
Created a new DOS (MBR) disklabel with disk identifier 0x857a78a6.

Orden (m para obtener ayuda): n
Tipo de partición
  p  primaria (0 primaria(s), 0 extendida(s), 4 libre(s))
  e  extendida (contenedor para particiones lógicas)
Seleccionar (valor predeterminado p): p
Número de partición (1-4, valor predeterminado 1):
Primer sector (2048-20963327, valor predeterminado 2048):
Último sector, +/-sectores o +/-tamaño{K,M,G,T,P} (2048-20963327, valor predeterminado 20963327):

Crea una nueva partición 1 de tipo 'Linux' y de tamaño 10 GiB.

Orden (m para obtener ayuda): w
Se ha modificado la tabla de particiones.
Llamando a ioctl() para volver a leer la tabla de particiones.
Se están sincronizando los discos.

oscarsanabria@debian:~$

```

- sudo mkfs.btrfs /dev/sda1

```

oscarsanabria@debian: ~
oscarsanabria@debian: ~ 211x46
oscarsanabria@debian:~$ lsblk -f
NAME                FSTYPE      FSVER      LABEL UUID                                 FSAVAIL FSUSE% MOUNTPOINTS
loop0
loop1
loop2
loop3
loop4
loop5
loop6
sda
└─sda1
nvme0n1
├─nvme0n1p1          vfat        FAT32      E4D7-CBA6                                480,2M   1% /boot/efi
├─nvme0n1p2          LVM2_member LVM2 001    cmiyxt-Z0Dd-1wC0-EK5K-pGY0-iyzE-pt166a
│   └─vg-01-Raiz     ext4        1.0       3892e1e0-53a0-4b31-b541-9592b1a1af99    125,9G   26% /
│       └─vg-01-Var  ext4        1.0       20f8045f-58f6-4460-ad9a-fc3ecc2ccf26    44,7G    54% /var
│           └─vg-01-Home ext4        1.0       05187b6e-cb01-41cb-9116-164ae394afc8    49,9G    66% /home
oscarsanabria@debian:~$

```

```

oscarsanabria@debian: ~
oscarsanabria@debian: ~ 211x46
oscarsanabria@debian:~$ sudo mkfs.btrfs /dev/sda1
btrfs-progs v6.2
See http://btrfs.wiki.kernel.org for more information.

NOTE: several default settings have changed in version 5.15, please make sure
this does not affect your deployments:
- DUP for metadata (-m dup)
- enabled no-holes (-O no-holes)
- enabled free-space-tree (-R free-space-tree)

Label:                (null)
UUID:                 aaa8aa93-aa1d-4c69-9048-7f5e9101bdf9
Node size:            16384
Sector size:          4096
Filesystem size:     10.00GiB
Block group profiles:
  Data:               single           8.00MiB
  Metadata:           DUP              256.00MiB
  System:             DUP              8.00MiB
SSD detected:         no
Zoned device:         no
Incompat features:    extref, skinny-metadata, no-holes
Runtime features:    free-space-tree
Checksum:             crc32c
Number of devices:    1
Devices:
  ID     SIZE  PATH
  1     10.00GiB /dev/sda1

```

- sudo mkdir /home/oscarsanabria/Escritorio/discocompartido

```

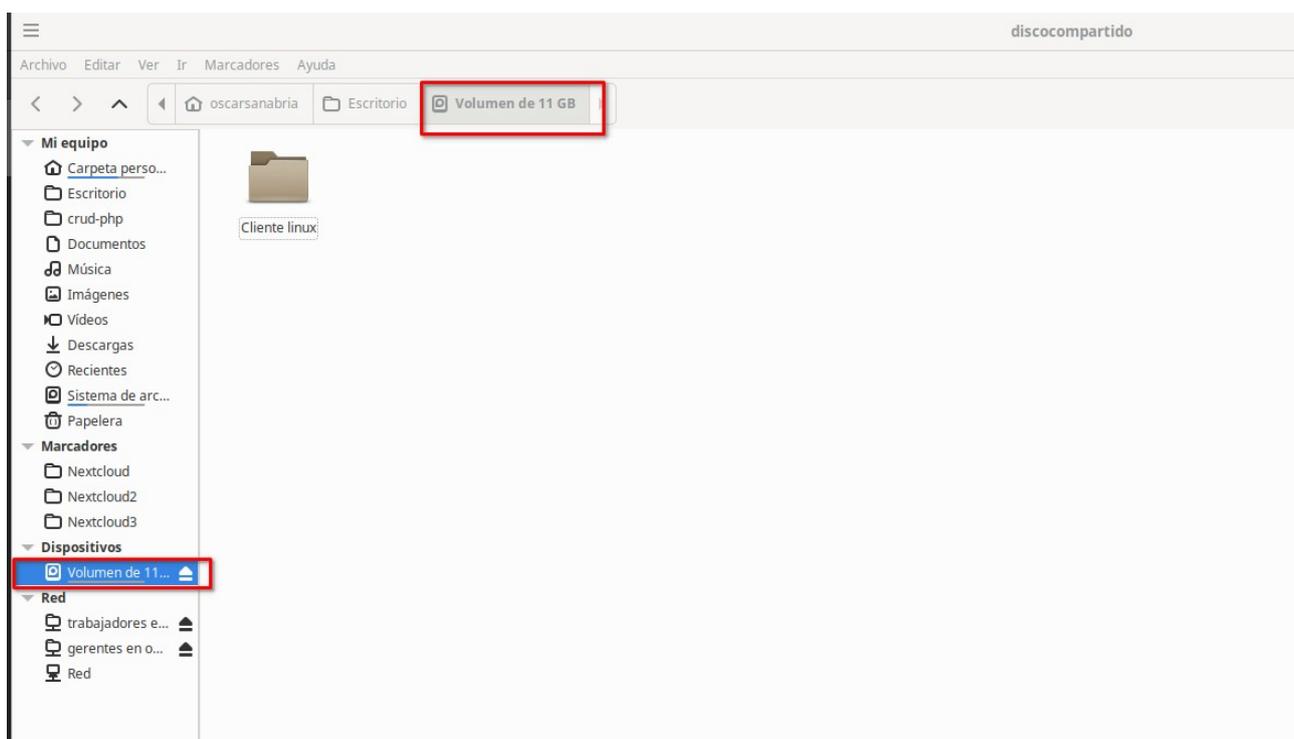
oscarsanabria@debian:~$ mkdir /home/oscarsanabria/Escritorio/discocompartido
oscarsanabria@debian:~$ ls /home/oscarsanabria/Escritorio/
ADSORAFa                datos.txt
APWjosedomingo          debian-oscar.csr
ataque.odt              discocompartido
ataque.py               ans.png
backupnext.sql          Dockerfile
BBDD                    document.pdf

```

- sudo mount /dev/sda1 /home/oscarsanabria/Escritorio/discocompartido

```
oscarsanabria@debian:~$ sudo mount /dev/sda1 /home/oscarsanabria/Escritorio/discocompartido
oscarsanabria@debian:~$ lsblk -f
bash: lsblk: orden no encontrada
oscarsanabria@debian:~$ lsblk -f
NAME                FSTYPE     FSVER     LABEL UUID                                FSAVAIL FSUSE% MOUNTPOINTS
loop0                vfat       FAT32     E4D7-CBA6                                0       100% /snap/core18/2812
loop1                vfat       FAT32     E4D7-CBA6                                0       100% /snap/bare/5
loop2                vfat       FAT32     E4D7-CBA6                                0       100% /snap/core18/2823
loop3                vfat       FAT32     E4D7-CBA6                                0       100% /snap/gtk-common-themes/1535
loop4                vfat       FAT32     E4D7-CBA6                                0       100% /snap/kde-frameworks-5-core18/35
loop5                vfat       FAT32     E4D7-CBA6                                0       100% /snap/snapd/21759
loop6                vfat       FAT32     E4D7-CBA6                                0       100% /snap/snapd/21465
sda
└─sda1               btrfs     3.19      aaa8aa93-aa1d-4c69-9048-7f5e9101bdf9    9,5G    0% /home/oscarsanabria/Escritorio/discocompartido
nvme0n1
├─nvme0n1p1         vfat       FAT32     E4D7-CBA6                                480,2M  1% /boot/efi
├─nvme0n1p2         LVM2_member LVM2 001    cmiyxt-Z0Dd-1wc0-EK5K-pGY0-iyzE-pt166a
│   └─vg--01-Raiz   ext4       1.0      3892e1e0-53a0-4b31-b541-9592b1a1af99    125,9G  26% /
│   └─vg--01-Var   ext4       1.0      20f8045f-58f6-4460-ad9a-fc3ecc2ccf26    44,7G  54% /var
│   └─vg--01-Home  ext4       1.0      05187b6e-cb01-41cb-9116-164ae394afc8    49,9G  66% /home
oscarsanabria@debian:~$
```

Como podemos ver ya tenemos nuestro disco montando y listo para utilizar.



Con esto no nos aseguramos de que nuestro disco se monte automáticamente cada cierto reinicio del equipo. Para lograr que se monte automáticamente este disco duro compartido por el servidor, debemos crear una unidad de systemd. Esta se encargará de montar el volumen cada vez que reiniciemos nuestro servidor OMV.

Creamos la unidad con el siguiente comando **sudo nano /etc/systemd/system/home-oscarsanabria-Escritorio-discocompartido.mount**

```

oscarsanabria@debian: ~
GNU nano 7.2 /etc/systemd/system/iscsi-tgtlun0.mount *
[Unit]
Description=Unidad de montaje para tgt
After=open-iscsi.service

[Mount]
What=/dev/sda1
Where=/home/oscarsanabria/Escritorio/discocompartido
Type=btrfs
Options=defaults

[Install]
WantedBy=multi-user.target

```

Reiniciamos los servicios.

- **sudo systemctl daemon-reload**
- **sudo systemctl enable home-oscarsanabria-Escritorio-discocompartido.mount**
- **sudo systemctl start home-oscarsanabria-Escritorio-discocompartido.mount**
- **sudo systemctl status home-oscarsanabria-Escritorio-discocompartido.mount**

```

oscarsanabria@debian:~$ sudo systemctl daemon-reload
oscarsanabria@debian:~$ sudo systemctl enable home-oscarsanabria-Escritorio-discocompartido.mount
Created symlink /etc/systemd/system/multi-user.target.wants/home-oscarsanabria-Escritorio-discocompartido.mount - /etc/systemd/system/home-oscarsanabria-Escritorio-discocompartido.mount.
oscarsanabria@debian:~$ sudo systemctl start home-oscarsanabria-Escritorio-discocompartido.mount
oscarsanabria@debian:~$ sudo systemctl status home-oscarsanabria-Escritorio-discocompartido.mount
● home-oscarsanabria-Escritorio-discocompartido.mount - Unidad de montaje para tgt
   Loaded: loaded (/etc/systemd/system/home-oscarsanabria-Escritorio-discocompartido.mount; enabled; preset: enabled)
   Active: active (mounted) since Sun 2024-06-02 20:01:46 CEST; 26min ago
     Where: /home/oscarsanabria/Escritorio/discocompartido
    What: /dev/sda1
oscarsanabria@debian:~$

```

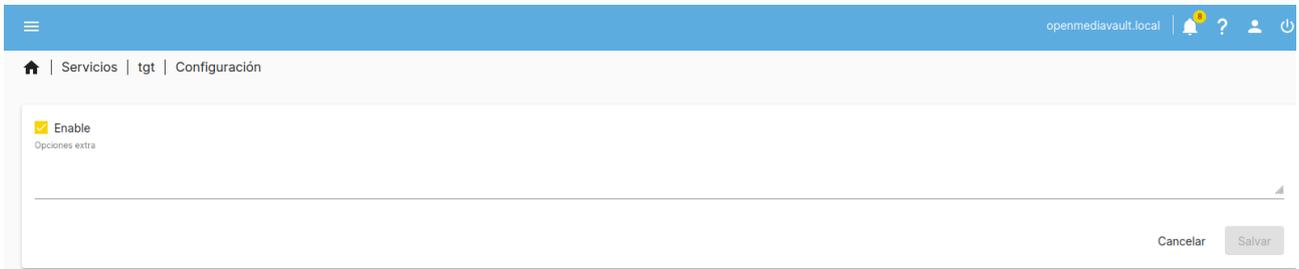
Como podemos ver, el servicio está en funcionamiento correctamente. Ahora tenemos el cliente Linux configurado correctamente para el uso de este disco compartido en nuestro servidor OMV.

4.6 Configuración de iSCSI con LVM – Cliente Windows.

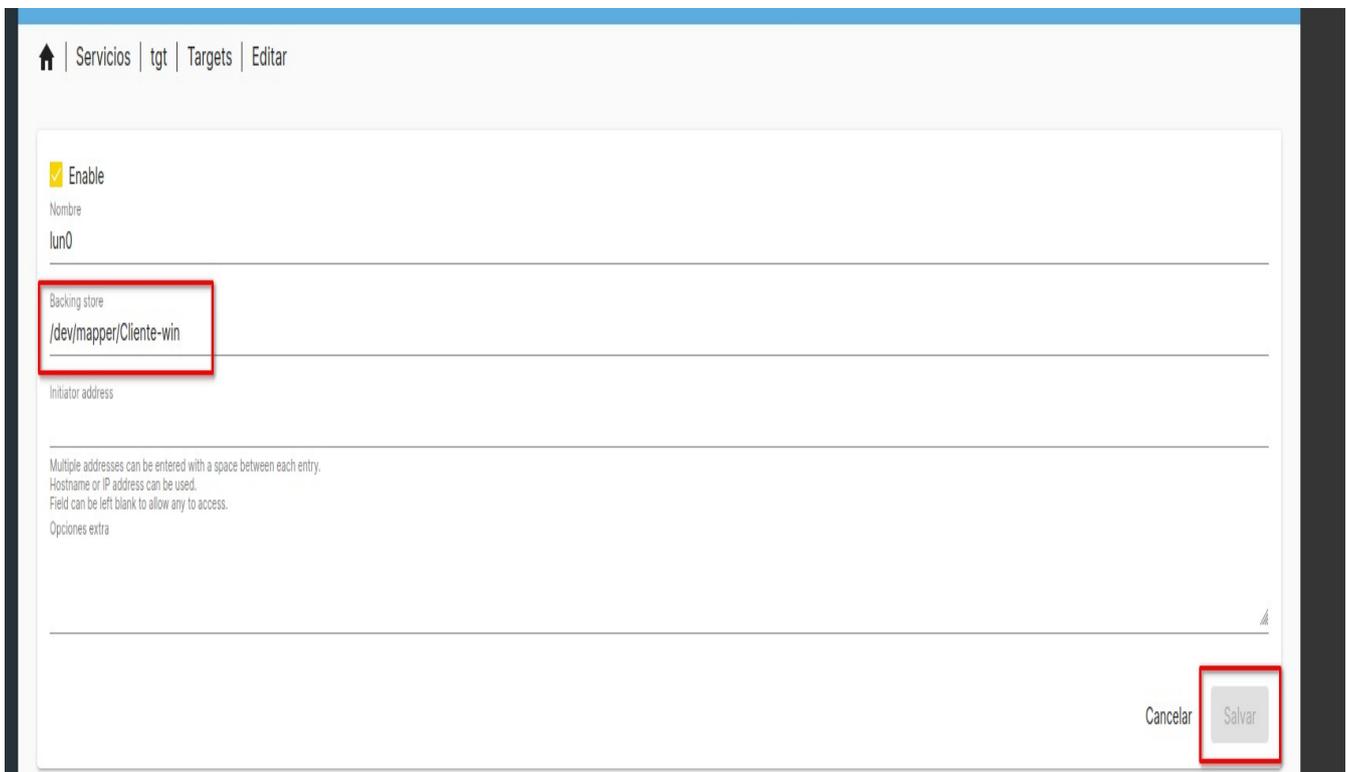
Lo primero que debemos hacer es instalar los siguientes plugins, los cuales nos permitirán realizar este tipo de configuraciones.

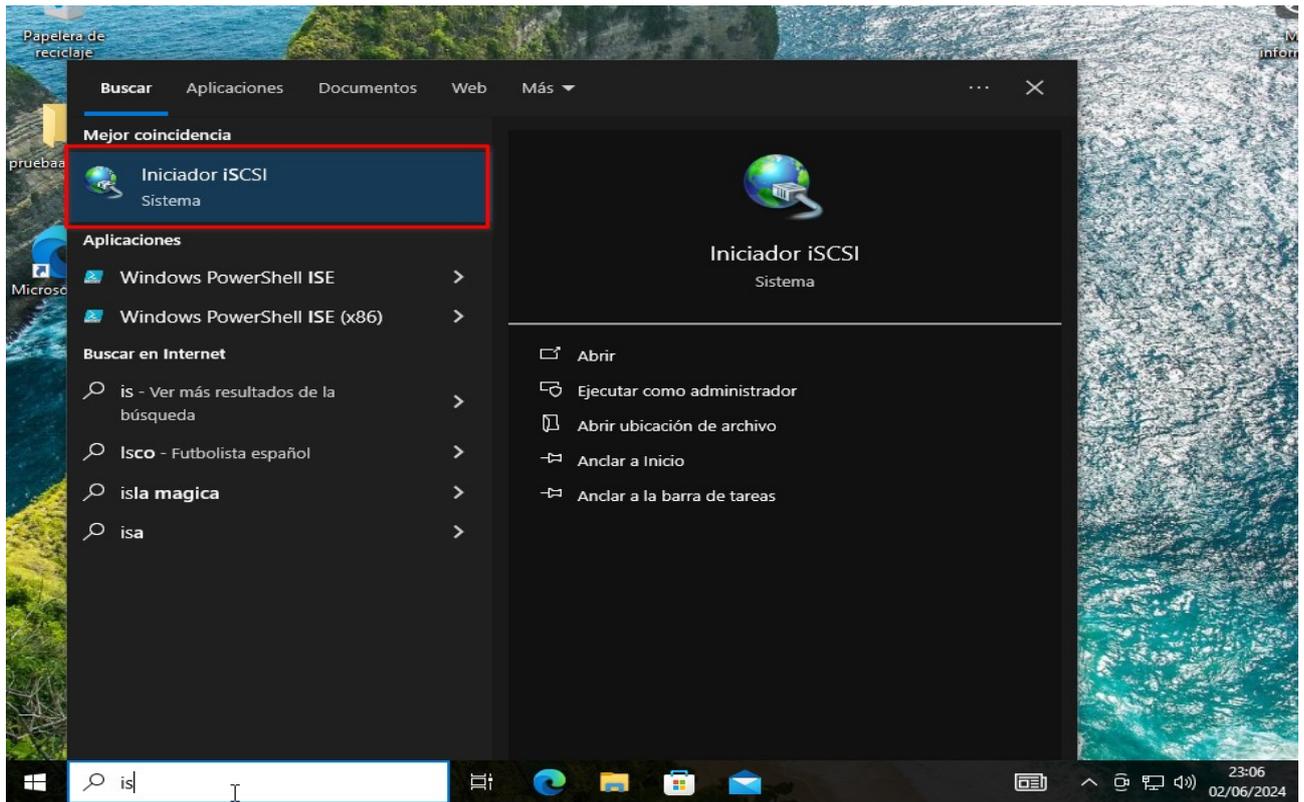


Una vez instalado el plugins, nos dirigimos al apartado de servicios donde encontraremos esta opción ya instalada. Para empezar, configuraremos primero el LVM en nuestro servidor OMV. Añadiré un disco de 10 GB, que compartiremos con el cliente Windows, y luego seguiremos con la creación de LVM como en los pasos anteriores para el cliente Linux. Habilitamos el servicio y creamos un target.

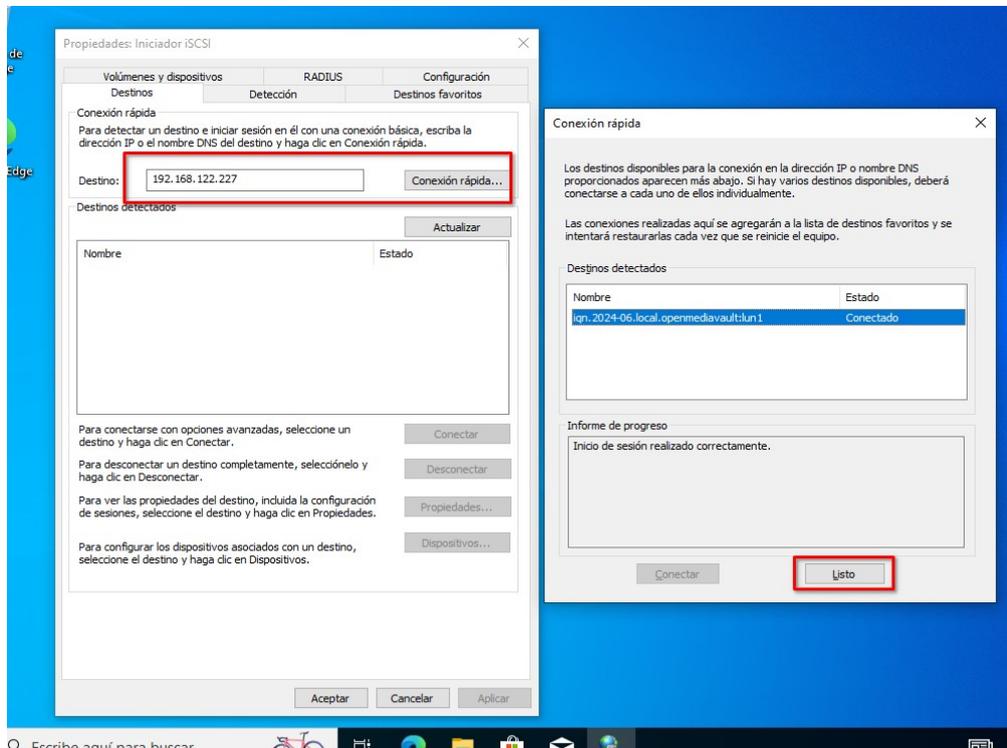


Ponemos el nombre como deseamos que se llame y el recurso a compartir, es decir, en mi caso el LVM que se creó anteriormente. En el recuadro que señalo en rojo, aquí podemos poner varias direcciones IP para los clientes donde queremos compartir los recursos. También podemos poner el nombre de la máquina y, en casos en los que no rellenemos este campo, automáticamente será visto por todos los PCs que estén en la misma red. En mi caso, lo dejaré en blanco. Pasamos a la configuración del cliente Windows.

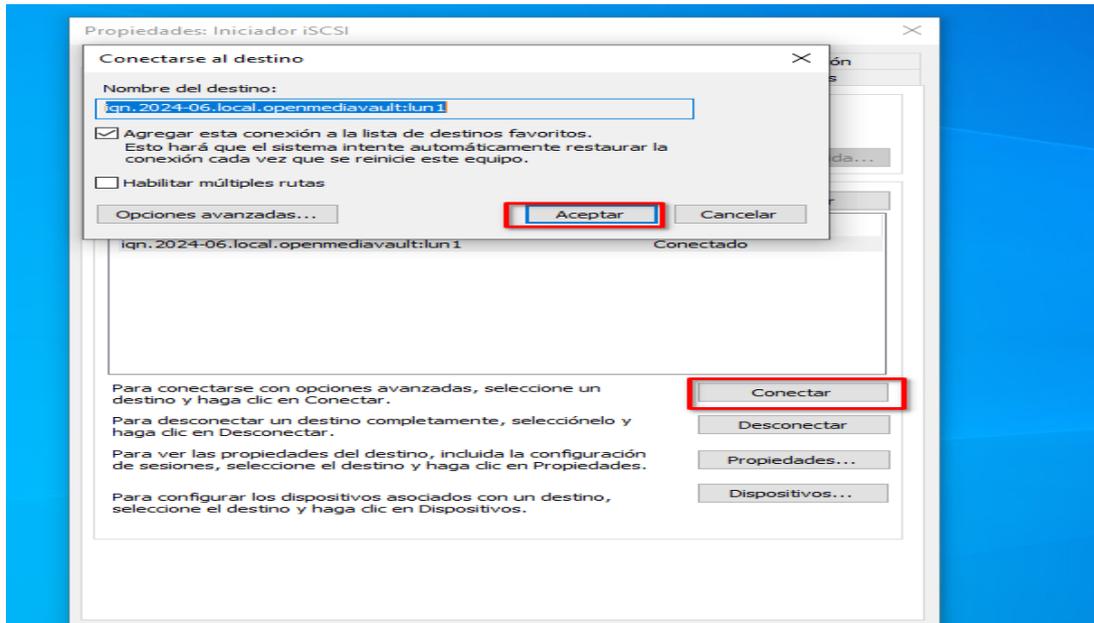




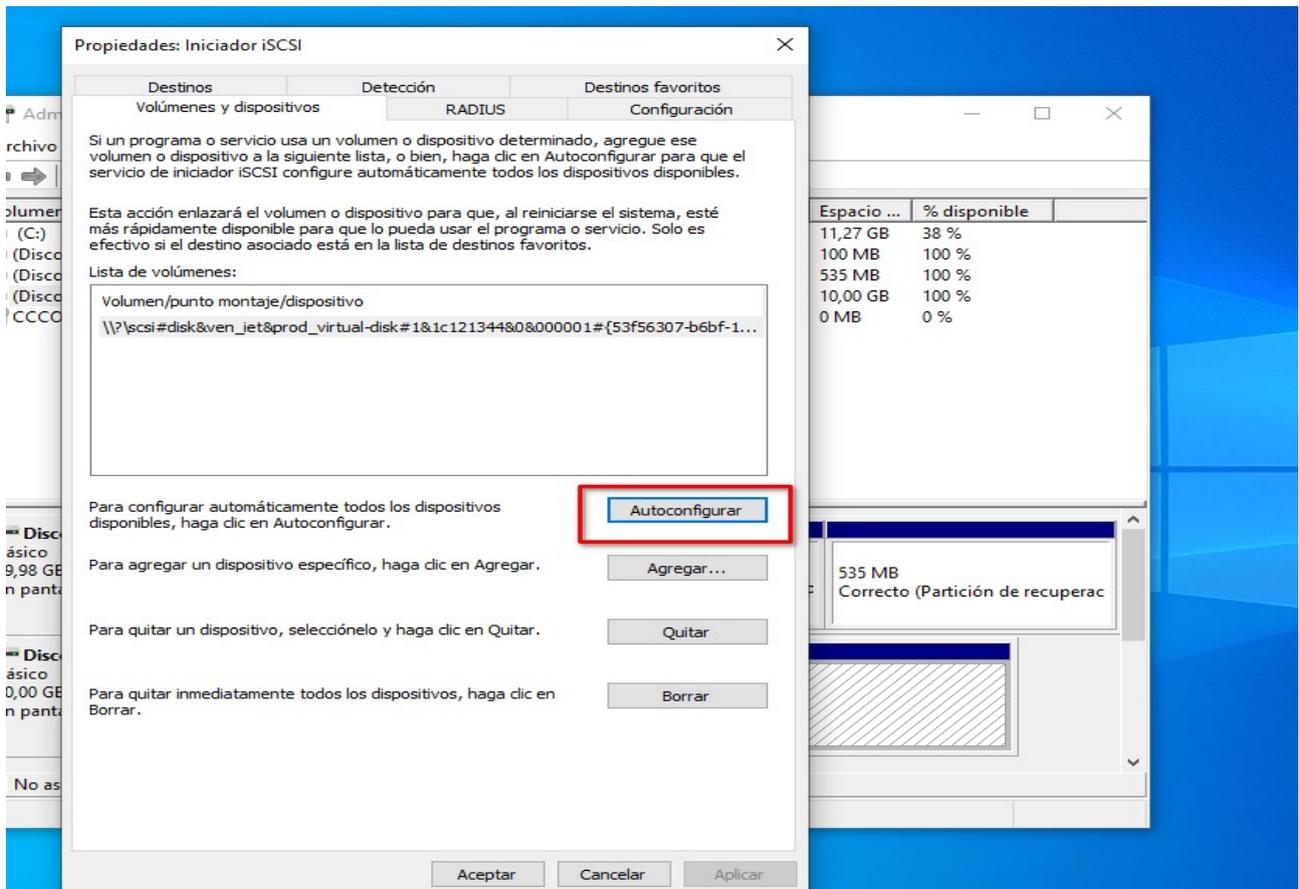
Introducimos la ip del servidor OMV para escanear target disponibles.



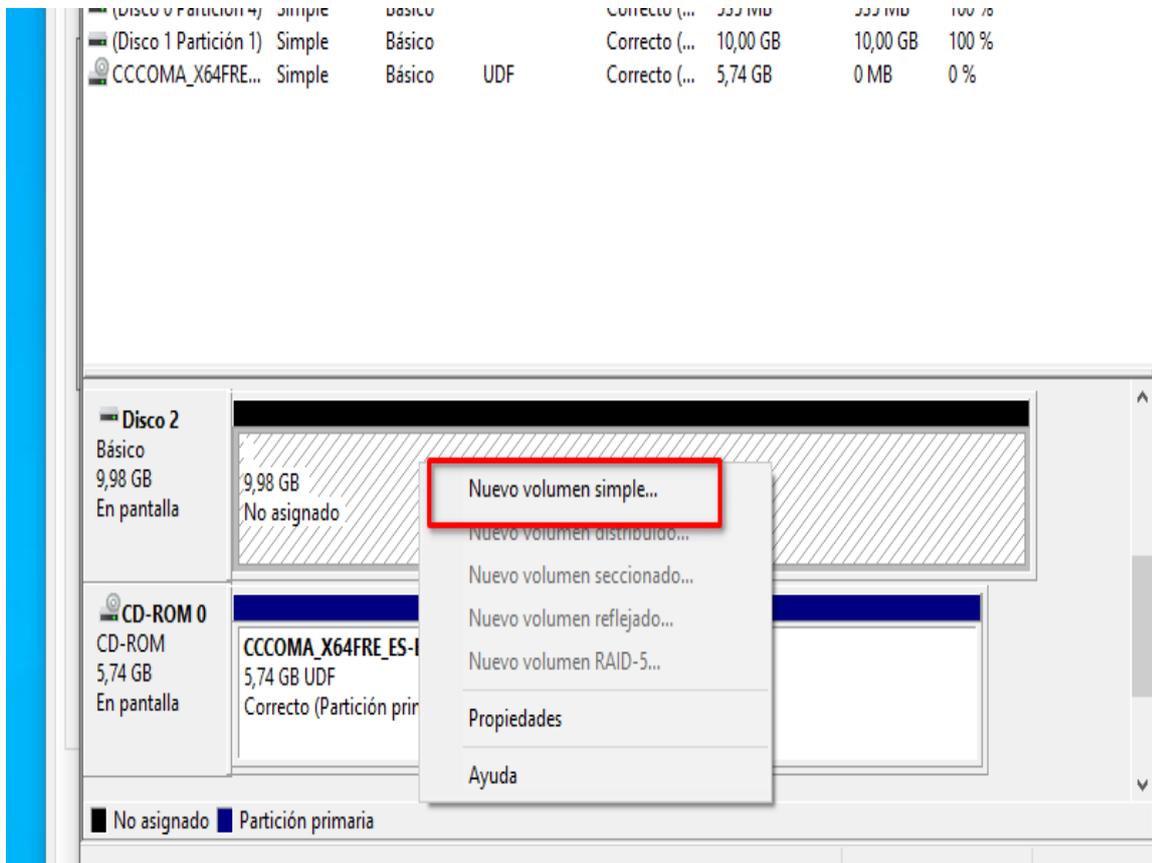
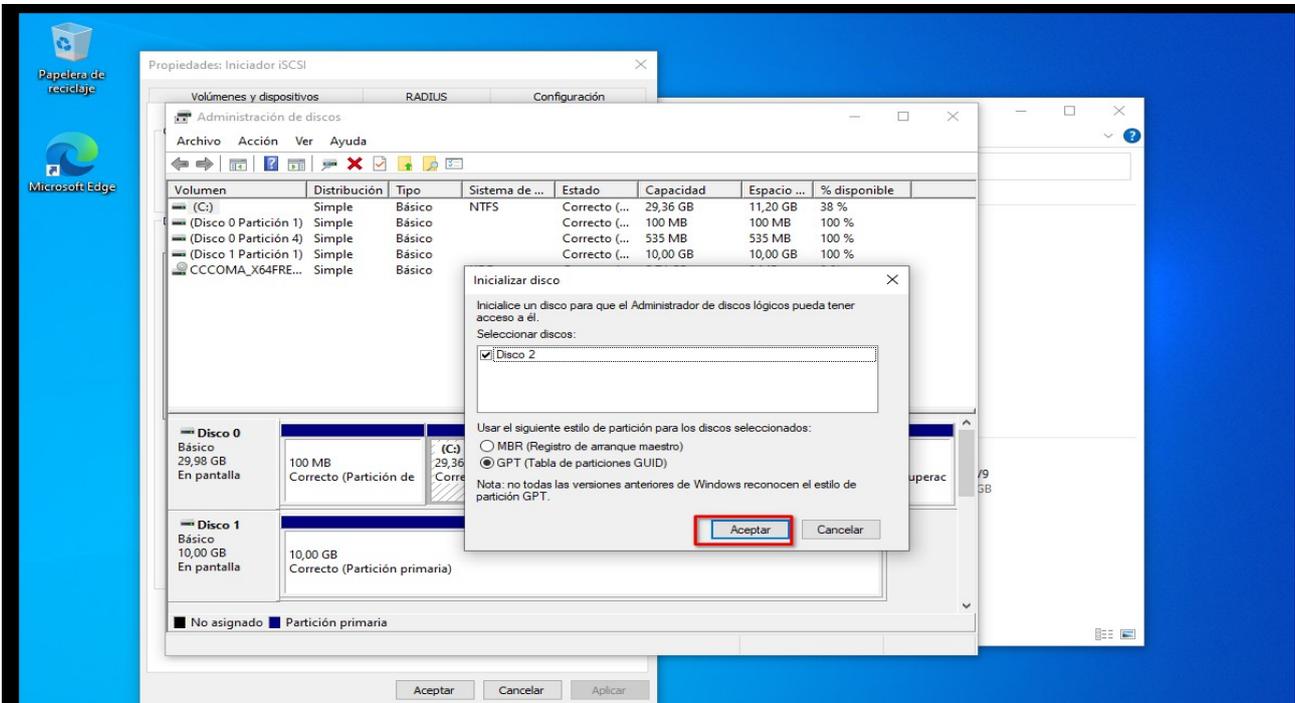
Conectamos con el target disponible

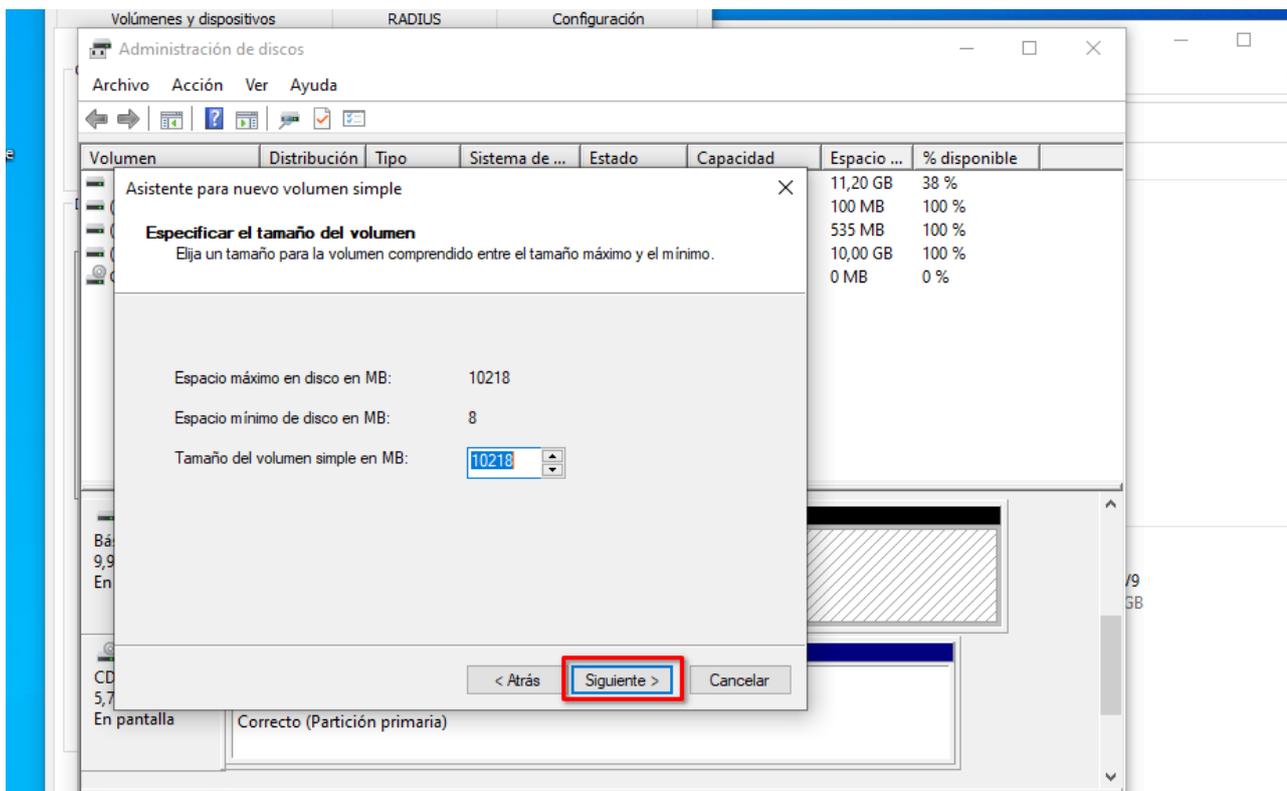
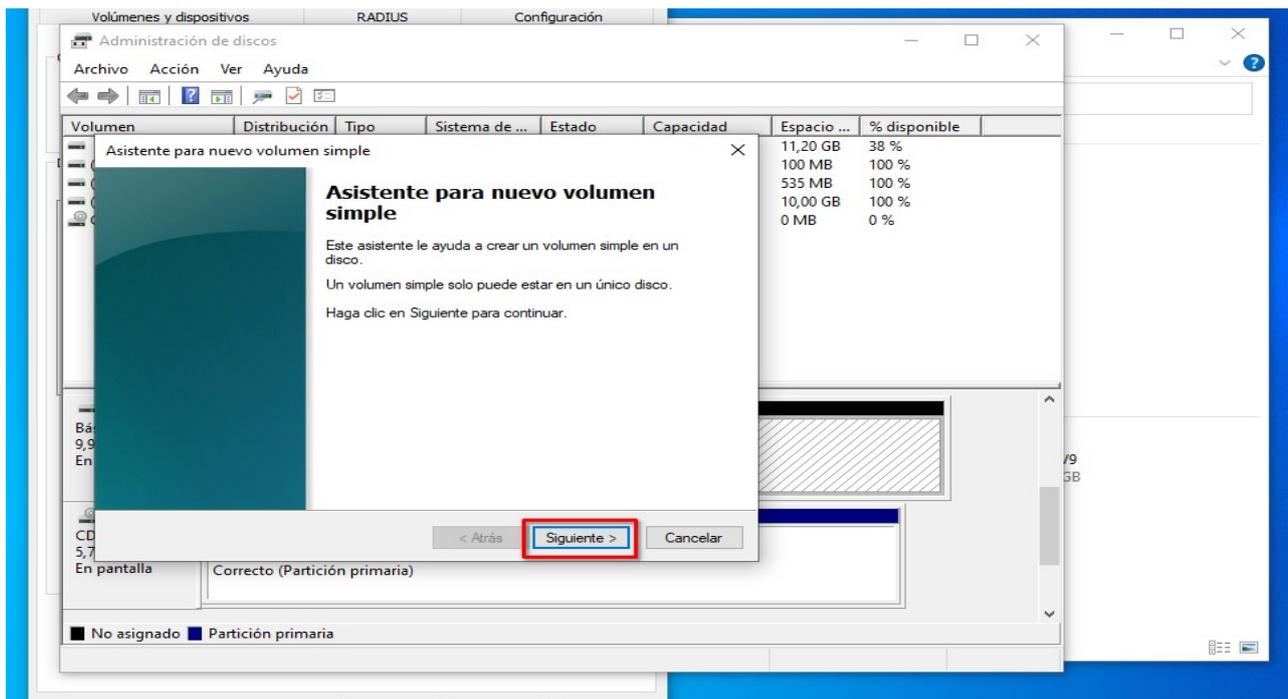


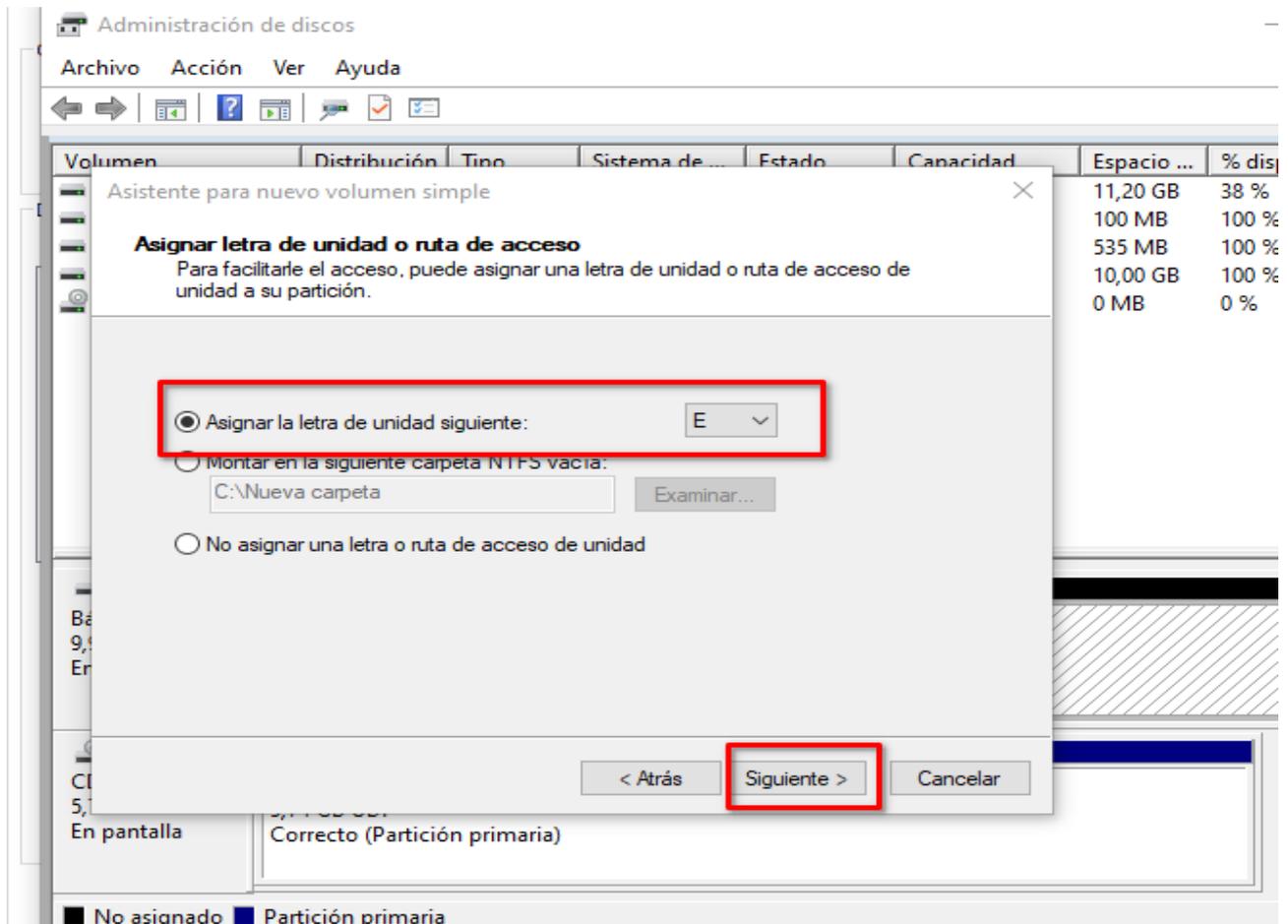
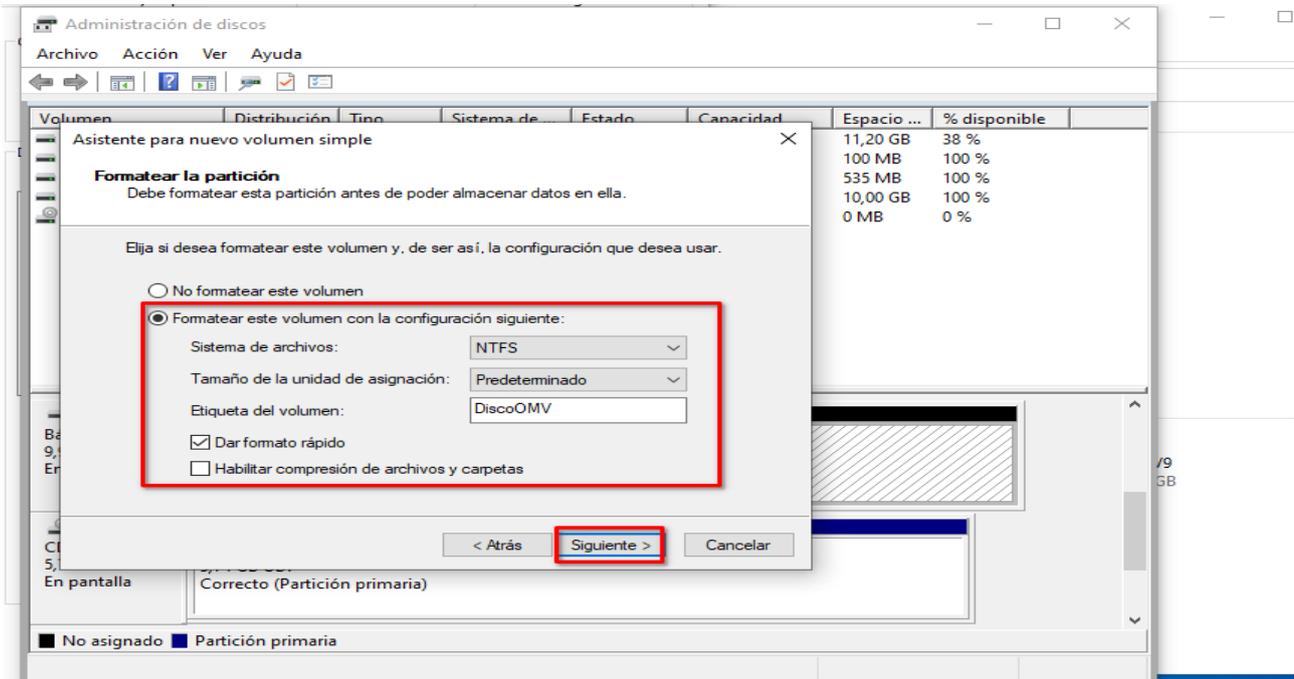
Auto configuramos el punto de montaje del propio disco compartido.

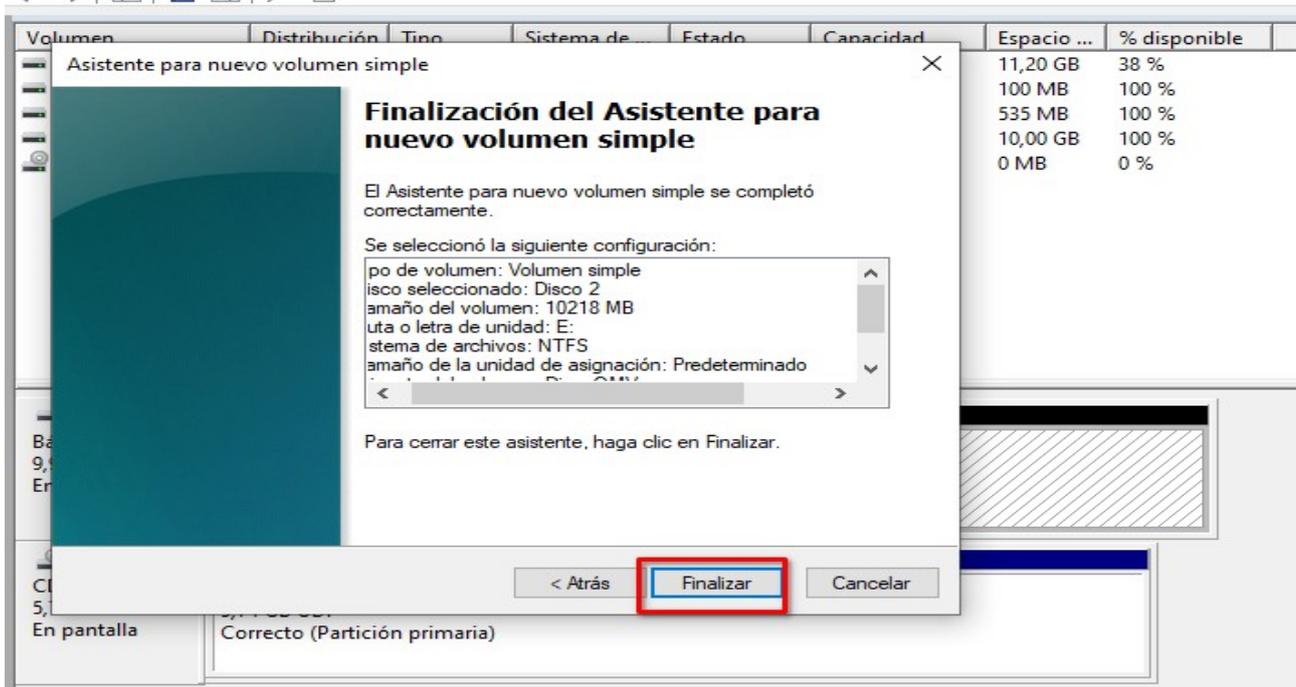


Nos dirigimos a la opción de partición de disco de Windows, donde nos indicará que disponemos de un nuevo disco duro en nuestro sistema. Para que este disco esté disponible, vamos a tener que realizar una serie de configuraciones. Vamos a formatear el nuevo volumen.

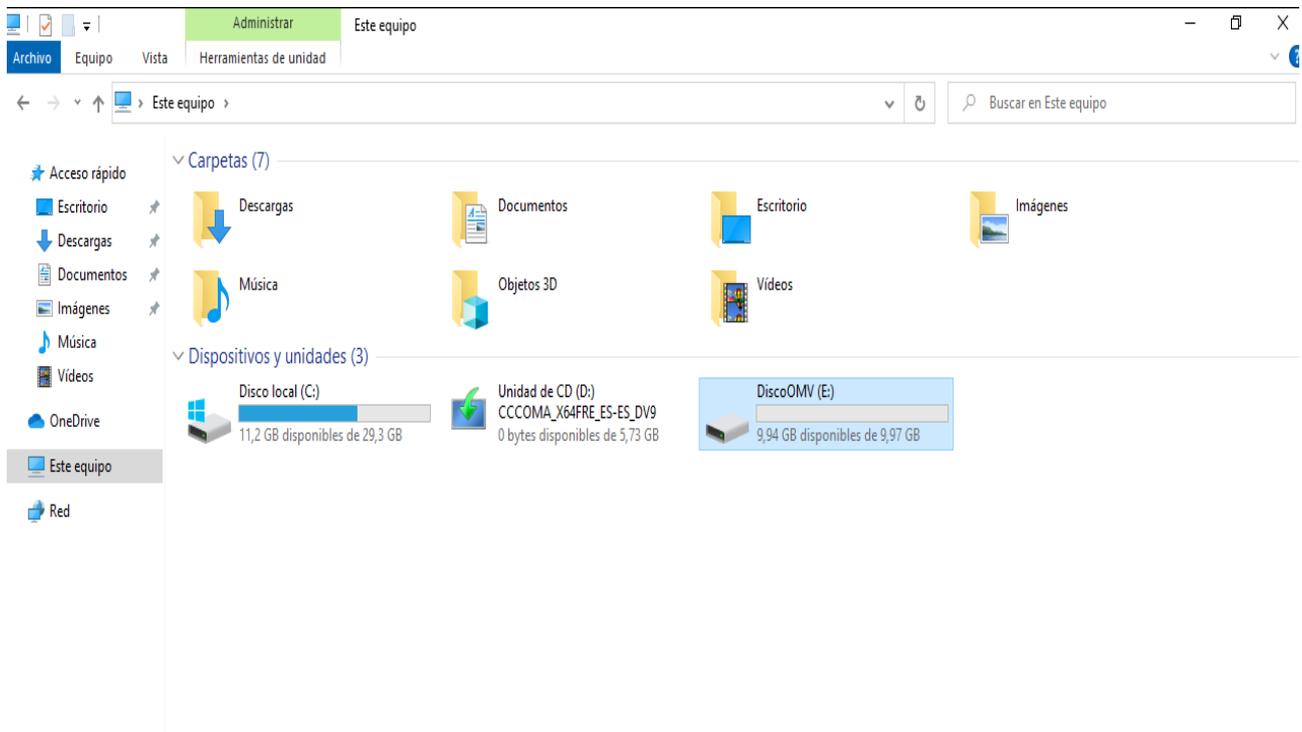


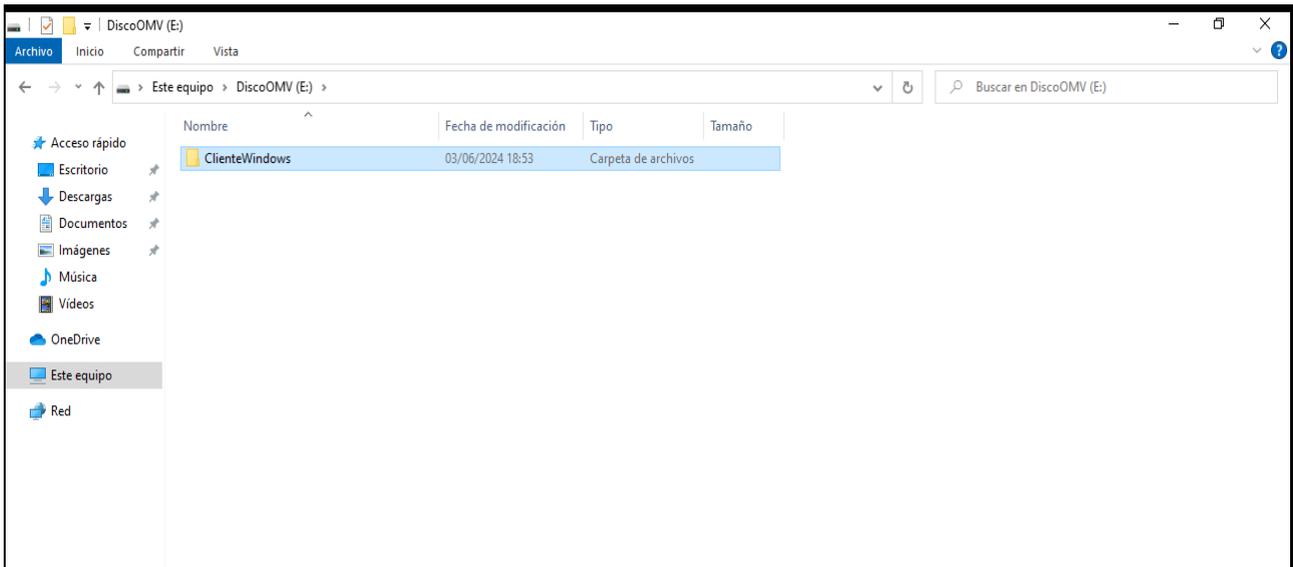






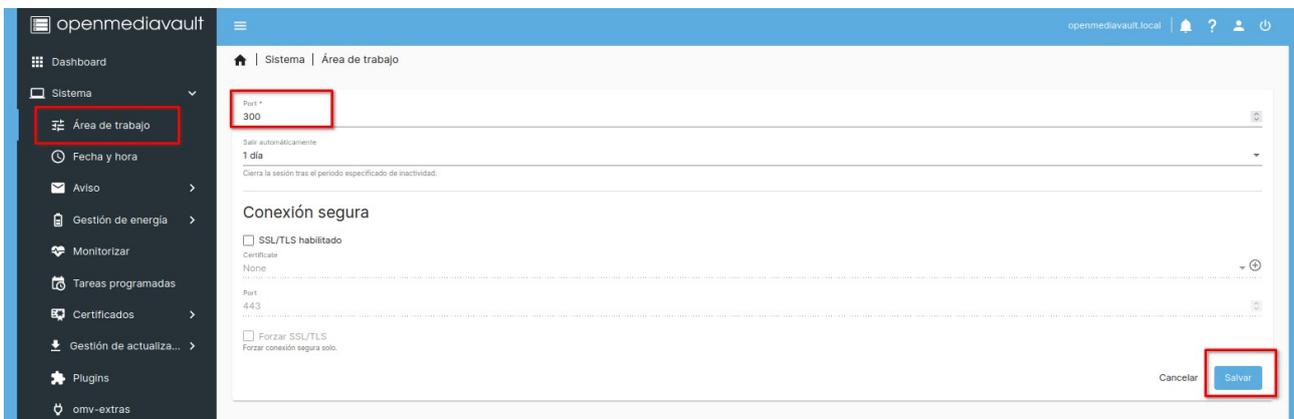
Podemos comprobar que tenemos disponible el disco del cliente Windows.



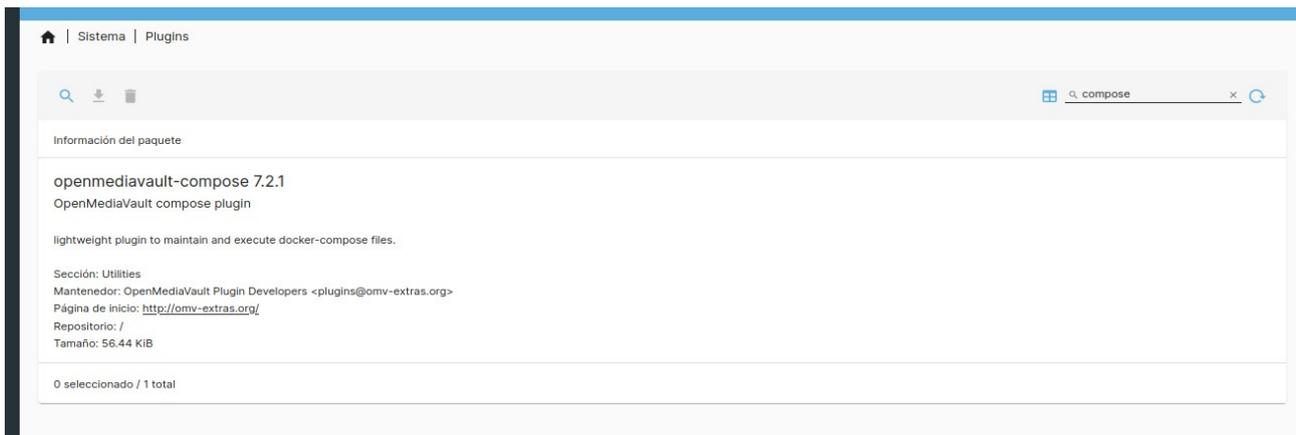


4.7 Configuración de Docker.

Antes de comenzar la instalación de Docker, vamos a dirigirnos a nuestro servidor y cambiar el puerto por donde está escuchando nuestra interfaz web. Esto es necesario porque al utilizar Docker, muchas aplicaciones pueden escuchar en el puerto 80, lo que podría causar conflictos.



Una vez realizado el cambio, guardamos y comenzamos la instalación. Para realizar esta configuración, debemos instalar el plugin de Docker.

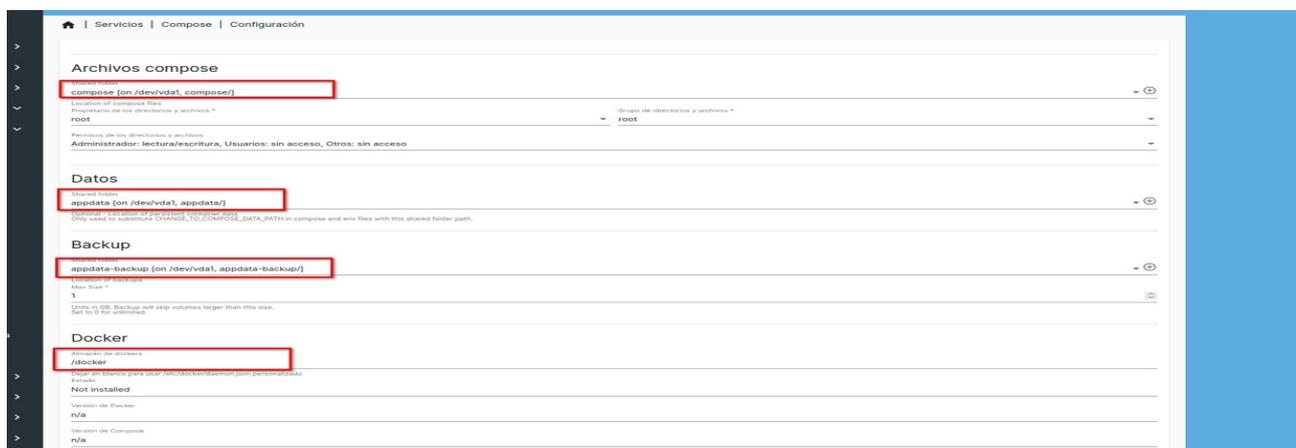


Una vez instalado, vamos a crear cuatro carpetas para configurar correctamente la instalación de Docker:

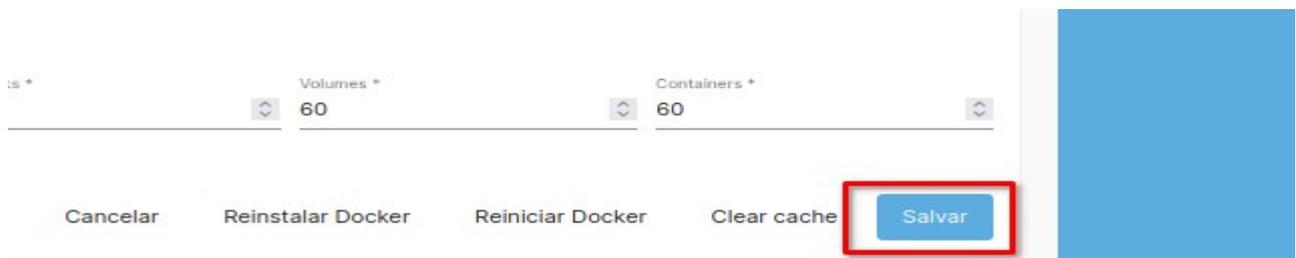
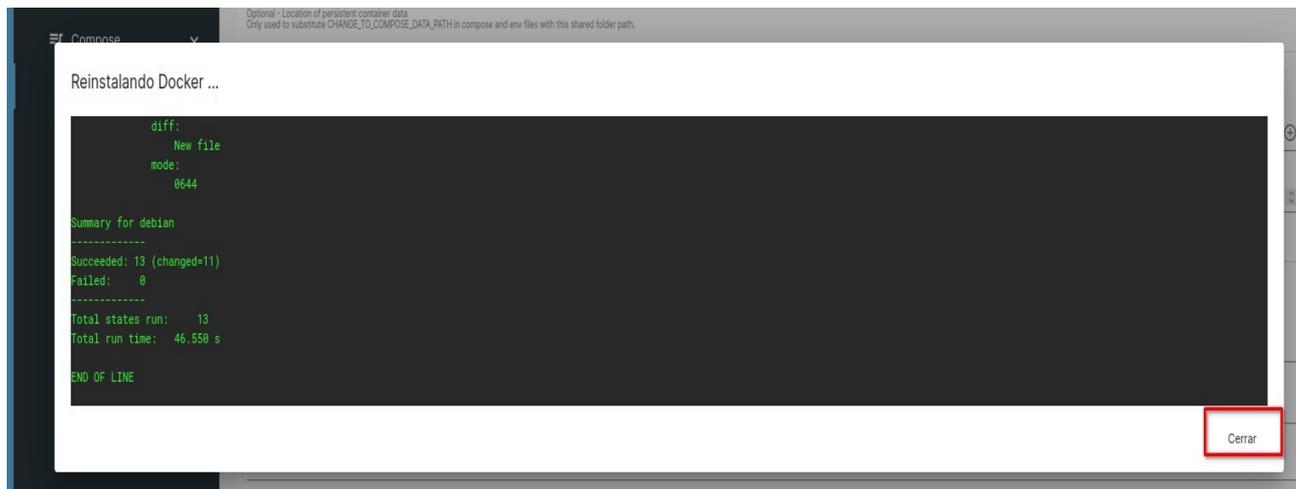
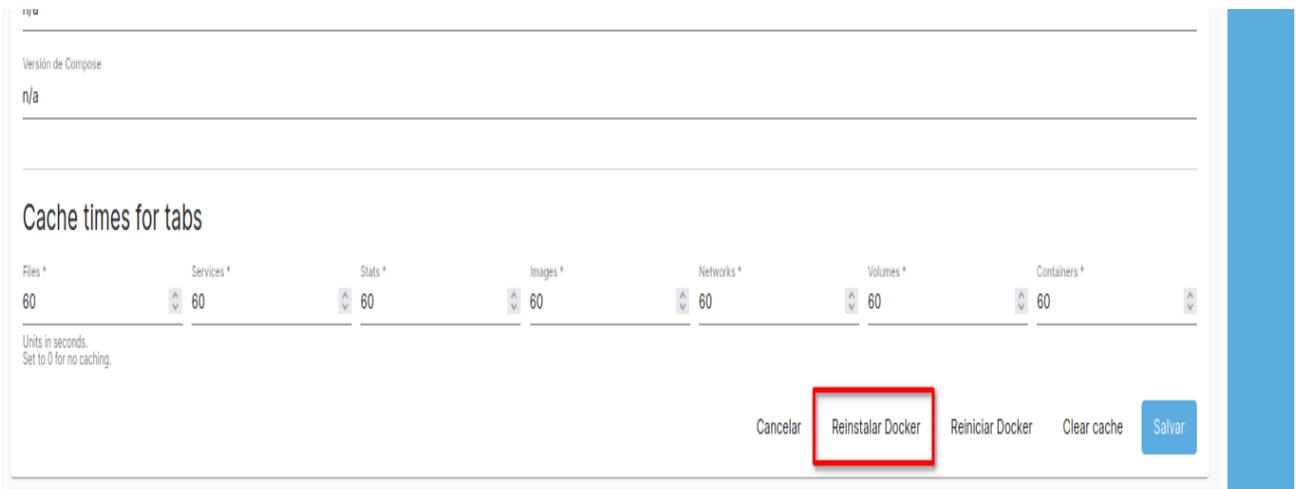
- **appdata**
- **appdata-backup**
- **compose**
- **docker**

appdata	/dev/vda1	appdata/	/appdata
appdata-backup	/dev/vda1	appdata-backup/	/appdata-backup
compose	/dev/vda1	compose/	/compose
docker	/dev/vda1	docker/	/docker

Nos dirigimos a la instalación de Docker y configuramos las carpetas creadas anteriormente.

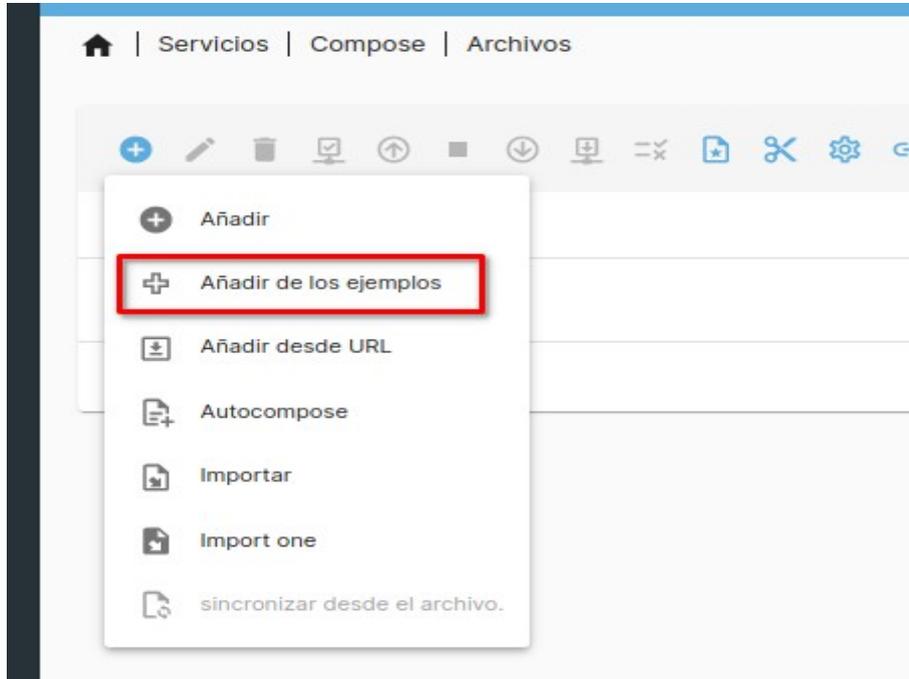


Tras la configuración de las carpetas, reinstalamos Docker.

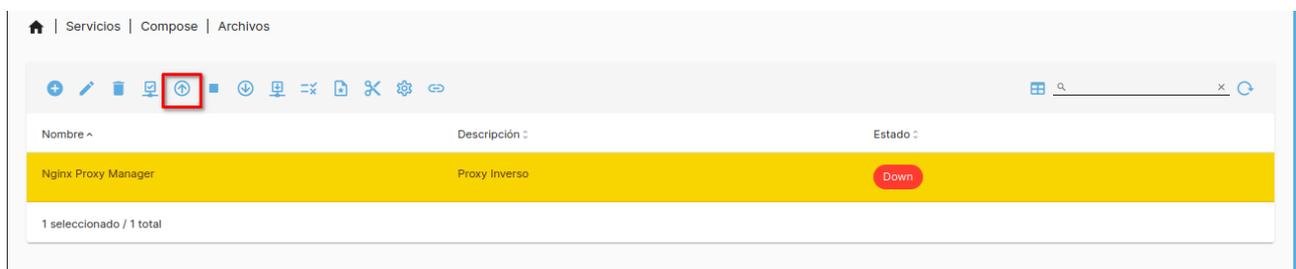


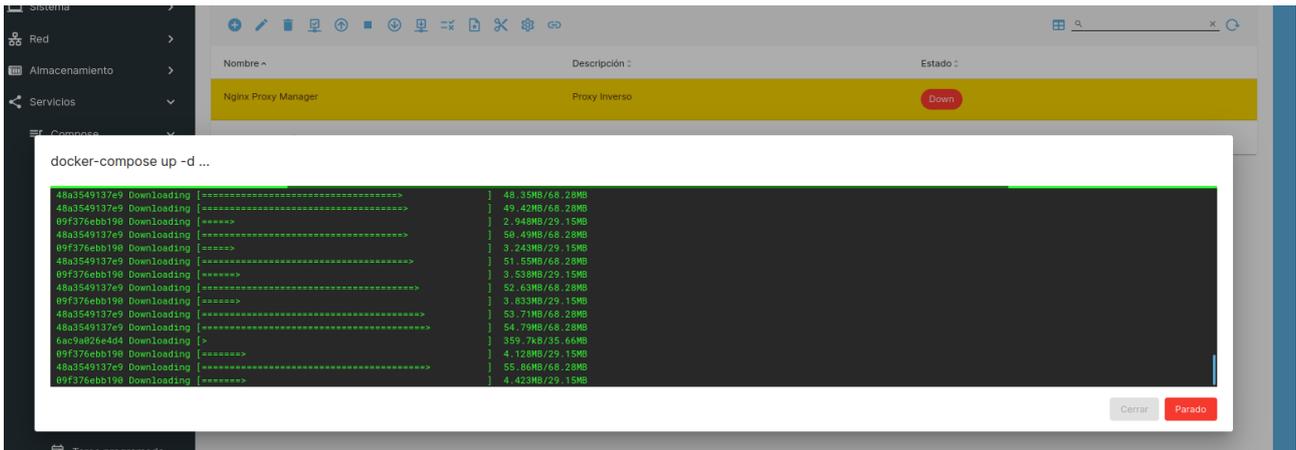
Tenemos Docker listo para su correcto funcionamiento. Vamos a realizar un ejemplo para comprobar que todo está en orden. Para ello, nos dirigimos al apartado de archivos y añadimos un ejemplo que proporciona OpenMediaVault por defecto; tienes muchas opciones para elegir. También podemos crear nuestros propios archivos Docker Compose.

Yo instalaré Nginx Proxy Manager, una herramienta que proporciona una interfaz fácil de usar para gestionar proxies inversos. Se ejecuta en Docker y permite redirigir el tráfico HTTP y HTTPS a diferentes servicios, además de manejar certificados TLS automáticamente. Es útil para exponer servicios internos de manera segura a través de Internet.



Levantamos el archivo docker-compose.

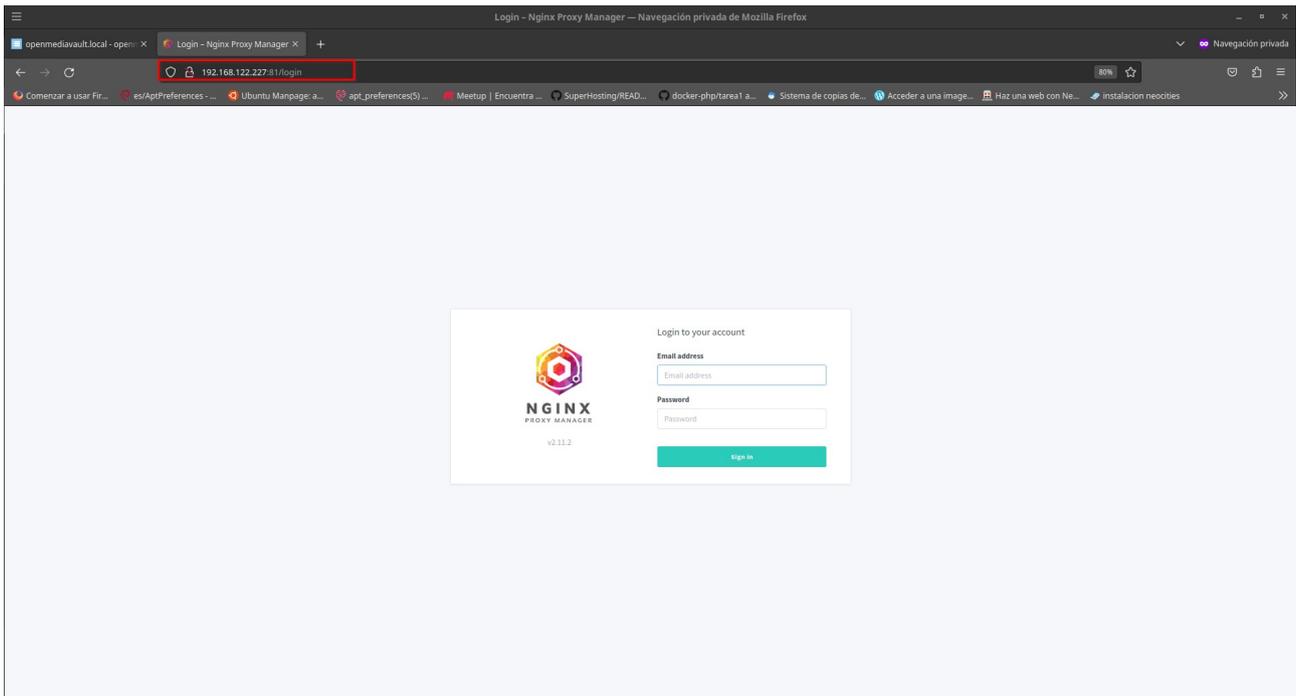




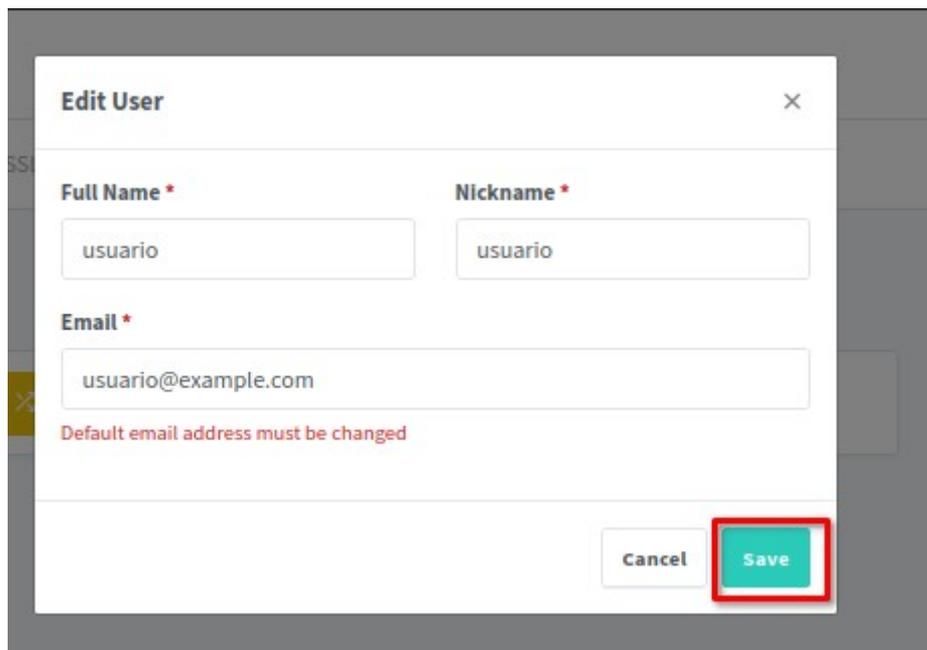
Si hacemos clic en el icono del lápiz, podemos editar el archivo docker-compose. Además, encontraremos las credenciales por defecto para acceder al servicio por primera vez.

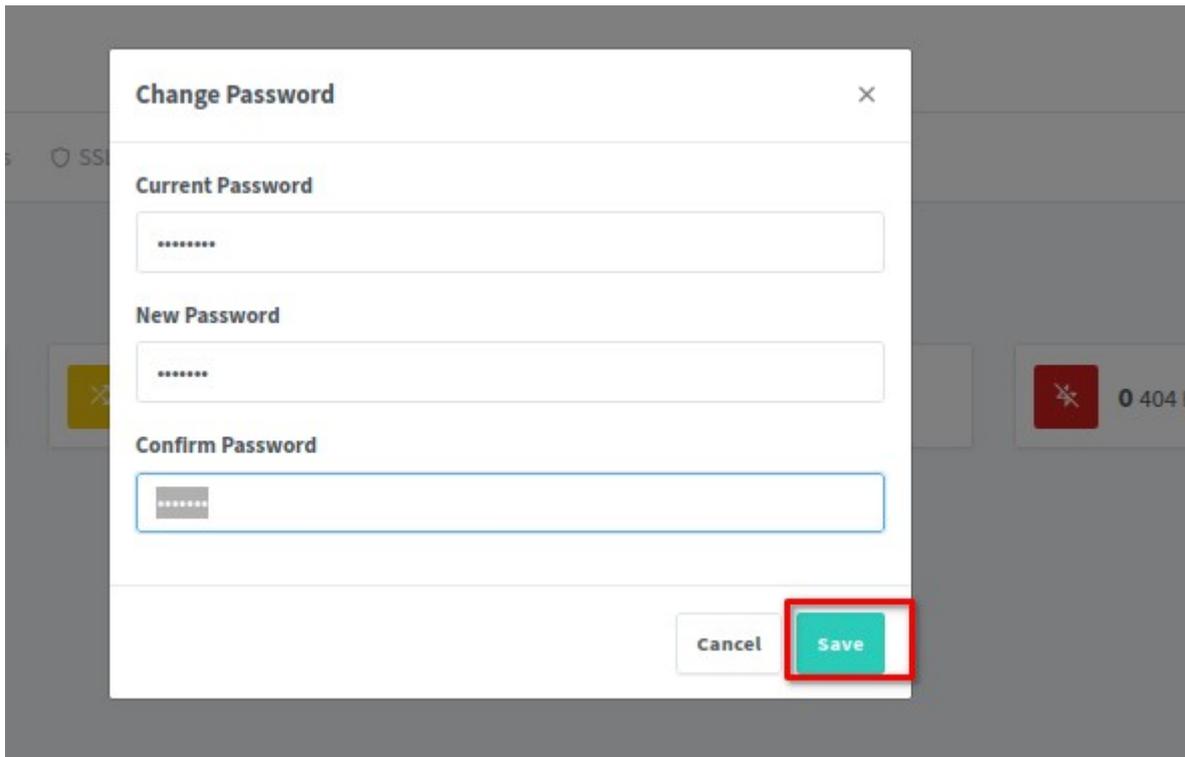


Accedemos al servicio que hemos levantado anteriormente.

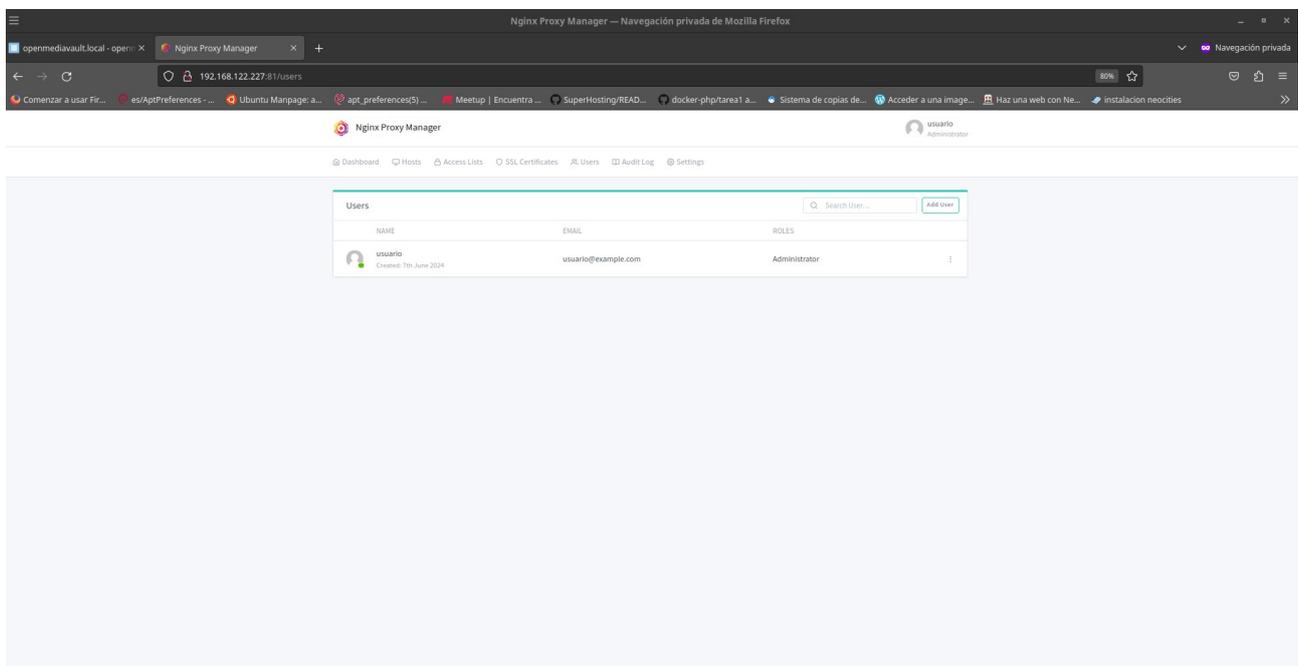


Una vez iniciado, te pedirá que cambies la información y la contraseña para mejorar la seguridad.





Ahora podemos ver que nuestro servicio está funcionando correctamente.



4.8 Backups con rclone

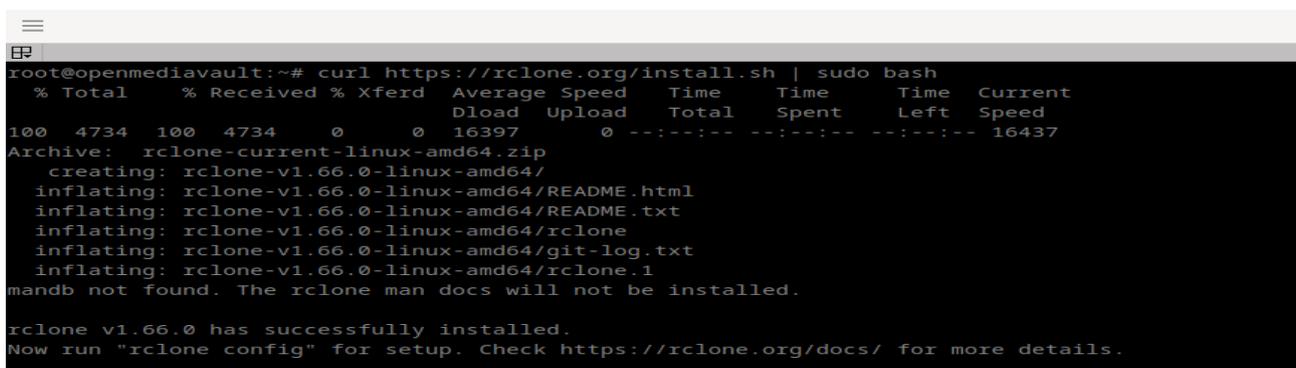
¿Que es rclone?

Rclone es una herramienta de línea de comandos diseñada para sincronizar archivos y directorios entre diferentes servicios de almacenamiento en la nube y sistemas de archivos locales. Con rclone, puedes transferir archivos de manera eficiente entre tu sistema local y diversos servicios en la nube, como Google Drive, Dropbox, Amazon S3 y muchos otros. Además de la sincronización de archivos, rclone también ofrece capacidades avanzadas como copia, movimiento, eliminación y listado de archivos, así como el cifrado de datos durante la transferencia y almacenamiento. Es una herramienta versátil y poderosa para gestionar tus archivos en múltiples ubicaciones de almacenamiento.

A continuación vamos a realizar un backup con rclone para realizar los siguientes pasos hacemos lo siguiente.

Instalamos rclone:

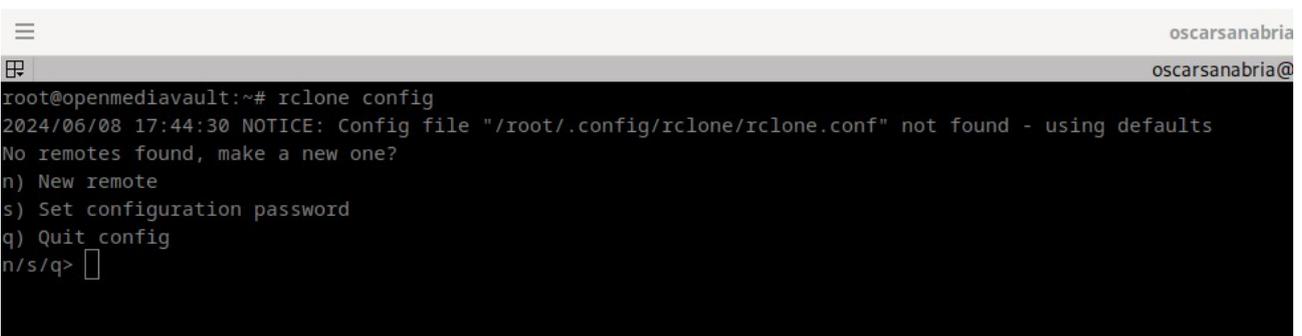
```
- curl https://rclone.org/install.sh | sudo bash
```



```
root@openmediavault:~# curl https://rclone.org/install.sh | sudo bash
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           %             %             Dload  Upload  Total   Spent    Left   Speed
100 4734    100 4734    0     0  16397      0  --:--:-- --:--:-- --:--:-- 16437
Archive:  rclone-current-linux-amd64.zip
  creating: rclone-v1.66.0-linux-amd64/
   inflating: rclone-v1.66.0-linux-amd64/README.html
   inflating: rclone-v1.66.0-linux-amd64/README.txt
   inflating: rclone-v1.66.0-linux-amd64/rclone
   inflating: rclone-v1.66.0-linux-amd64/git-log.txt
   inflating: rclone-v1.66.0-linux-amd64/rclone.1
mandb not found. The rclone man docs will not be installed.

rclone v1.66.0 has successfully installed.
Now run "rclone config" for setup. Check https://rclone.org/docs/ for more details.
```

Una vez instalado rclone, procedemos a configurarlo. Utilizamos el comando **rclone config** para iniciar la configuración.



```
root@openmediavault:~# rclone config
2024/06/08 17:44:30 NOTICE: Config file "/root/.config/rclone/rclone.conf" not found - using defaults
No remotes found, make a new one?
n) New remote
s) Set configuration password
q) Quit config
n/s/q> [ ]
```

Al configurar un nuevo remoto, seleccionamos la opción "n" para crear uno nuevo.

```

root@openmediavault:~# rclone config
2024/06/08 17:44:30 NOTICE: Config file "/root/.config/rclone/rclone.conf" not found - using default
No remotes found, make a new one?
n) New remote
s) Set configuration password
q) Quit config
n/s/q> n
    
```

Indicamos el nombre que deseamos asignar a nuestro remoto. Puedes elegir el nombre que prefieras.

```

Enter name for new remote.
name> cloudomv
    
```

Cuando avanzamos, nos encontramos con una lista de proveedores entre los cuales podemos elegir para configurar nuestro remoto. Esto dependerá de las preferencias personales. En mi caso, seleccionare Google Driver, ya que es el proveedor que utilizaremos para las pruebas.

```

Option Storage.
Type of storage to configure.
Choose a number from below, or type in your own value.
 1 / rFichier
   \ (fichier)
 2 / Akamai NetStorage
   \ (netstorage)
 3 / Alias for an existing remote
   \ (alias)
 4 / Amazon S3 Compliant Storage Providers including AWS, Alibaba, ArvanCloud, Ceph, ChinaMobile, Cloudflare, DigitalOcean, Dreamhost, GCS, HuaweiOBS, IBMCOS, IDrive, IONOS, LyveCloud, Leviia, Liara, Linode, Microsoft, Netase, Petabox, RackCorp, RClone, Scaleway, SeaweedFS, StackPath, Storj, Synology, TencentCOS, Wasabi, Qiniu and others
   \ (s3)
 5 / Backblaze B2
   \ (b2)
 6 / Better checksums for other remotes
   \ (hasher)
 7 / Box
   \ (box)
 8 / Cache a remote
   \ (cache)
 9 / Citrix Sharefile
   \ (sharefile)
10 / Combine several remotes into one
   \ (combine)
11 / Compress a remote
   \ (compress)
12 / Dropbox
   \ (dropbox)
13 / Encrypt/Decrypt a remote
   \ (crypt)
14 / Enterprise File Fabric
   \ (filefabric)
15 / FTP
   \ (ftp)
16 / Google Cloud Storage (this is not Google Drive)
   \ (google cloud storage)
17 / Google Drive
   \ (drive)
18 / Google Photos
   \ (google photos)
19 / HTTP
   \ (http)
20 / Hadoop distributed file system
   \ (hdfs)
21 / HDrive
    
```

En esta situación, podemos elegir el nivel de permisos que queremos otorgar a rclone. El nivel 1 es el más alto con acceso completo a todos los archivos y el nivel 5 es el más bajo con acceso solo a metadatos. En mi caso, para demostrar correctamente el programa, le daré el nivel máximo de permisos 1.

```

$ rclone config
Storage> 17
Option client_id.
Google Application Client Id.
Setting your own is recommended.
See https://rclone.org/drive/#making-your-own-client-id for how to create your own.
If you leave this blank, it will use an internal key which is low performance.
Enter a value. Press Enter to leave empty.
client_id>
Option client_secret.
OAuth Client Secret.
Leave blank normally.
Enter a value. Press Enter to leave empty.
client_secret>
Option scope.
Comma separated list of scopes that rclone should use when requesting access from drive.
Choose a number from below, or type in your own value.
Press Enter to leave empty.
1 / Full access all files, excluding Application Data Folder.
  \ (drive)
  \ (drive_readonly)
  / Access to files created by rclone only.
  | These are visible in the drive website.
  | File authorization is revoked when the user deauthorizes the app.
  \ (drive_file)
  / Allows read and write access to the Application Data folder.
  | This is not visible in the drive website.
  \ (drive_appfolder)
  / Allows read-only access to file metadata but
  | does not allow any access to read or download file content.
  \ (drive_metadata_readonly)
scope> 1

```

Después de seleccionar la opción "no" en la configuración avanzada y no tener un navegador web en el servidor OMV, rclone proporcionará un comando que debe ejecutarse en una máquina con rclone y un navegador web instalados, preferiblemente con la misma versión.

```

Option service_account_file.
Service Account Credentials JSON file path.
Leave blank normally.
Needed only if you want use SA instead of interactive login.
Leading `~` will be expanded in the file name as will environment variables such as `${RCLONE_CONFIG_DIR}`.
Enter a value. Press Enter to leave empty.
service_account_file>

Edit advanced config?
y) Yes
n) No (default)
y/n> n

Use web browser to automatically authenticate rclone with remote?
* Say Y if the machine running rclone has a web browser you can use
* Say N if running rclone on a (remote) machine without web browser access
If not sure try Y. If Y failed, try N.

y) Yes (default)
n) No
y/n> n

Option config_token.
For this to work, you will need rclone available on a machine that has
a web browser available.
For more help and alternate methods see: https://rclone.org/remote_setup/
Execute the following on the machine with the web browser (same rclone
version recommended):
rclone authorize "drive" "eyJkZXNjcmlwdGlvbiI6IiIsInNjb3BlIjoizHJpdmUifQ"
Then paste the result.
Enter a value.
config_token>

```

Una vez ejecutado el comando en tu propia máquina, se abrirá un navegador web donde podrás configurar tu cuenta de Google Drive y otorgar todos los permisos necesarios a rclone.

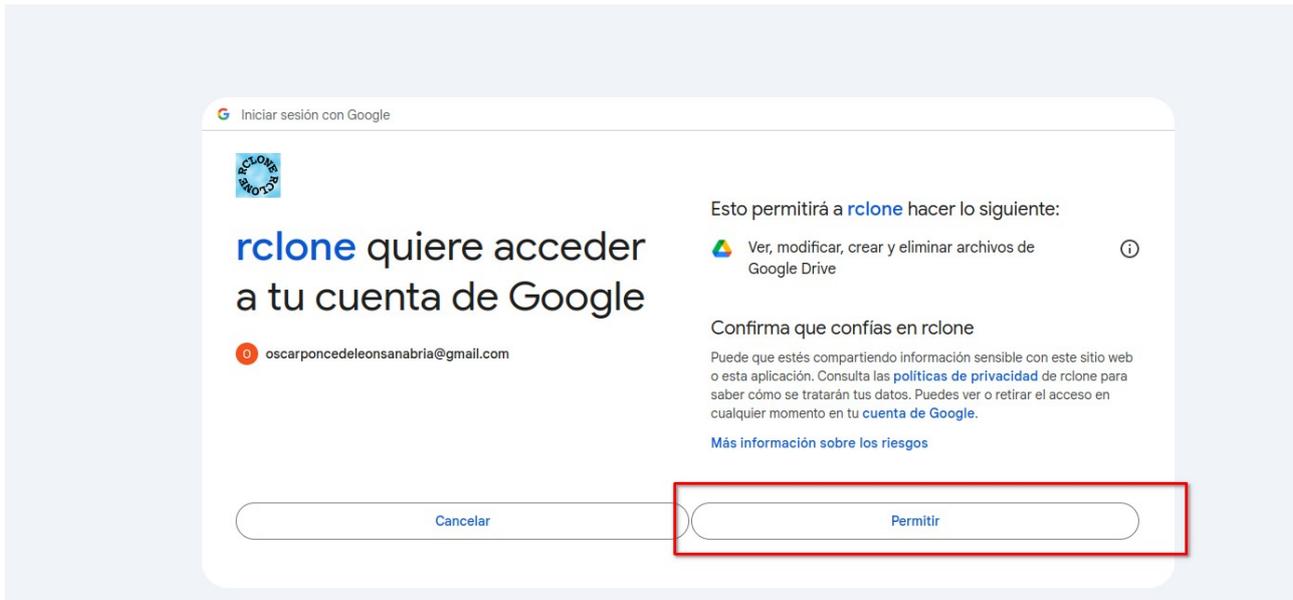
```
oscarosanabria@debian: ~  
root@debian:~# rclone authorize "drive" "eyJkZXNjcmldG1vbiI6IiIsInNjb3B1IjoizHJpdmUifQ"
```

```
oscarosanabria@debian: ~  
root@debian:~# rclone authorize "drive" "eyJkZXNjcmldG1vbiI6IiIsInNjb3B1IjoizHJpdmUifQ"  
2024/06/08 19:09:38 NOTICE: Config file "/root/.config/rclone/rclone.conf" not found - using defaults  
2024/06/08 19:09:38 NOTICE: If your browser doesn't open automatically go to the following link: http://127.0.0.1:53682/auth?state=vhcGq-Ss_0ox-nUF727dgc  
2024/06/08 19:09:38 NOTICE: Log in and authorize rclone for access  
2024/06/08 19:09:38 NOTICE: Waiting for code...
```

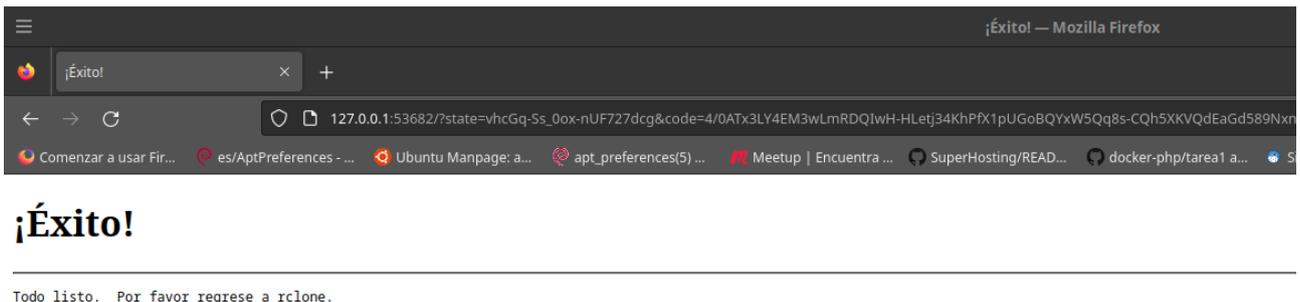
Seleccionamos la cuenta de Google que deseamos configurar para que se realicen las copias de seguridad.



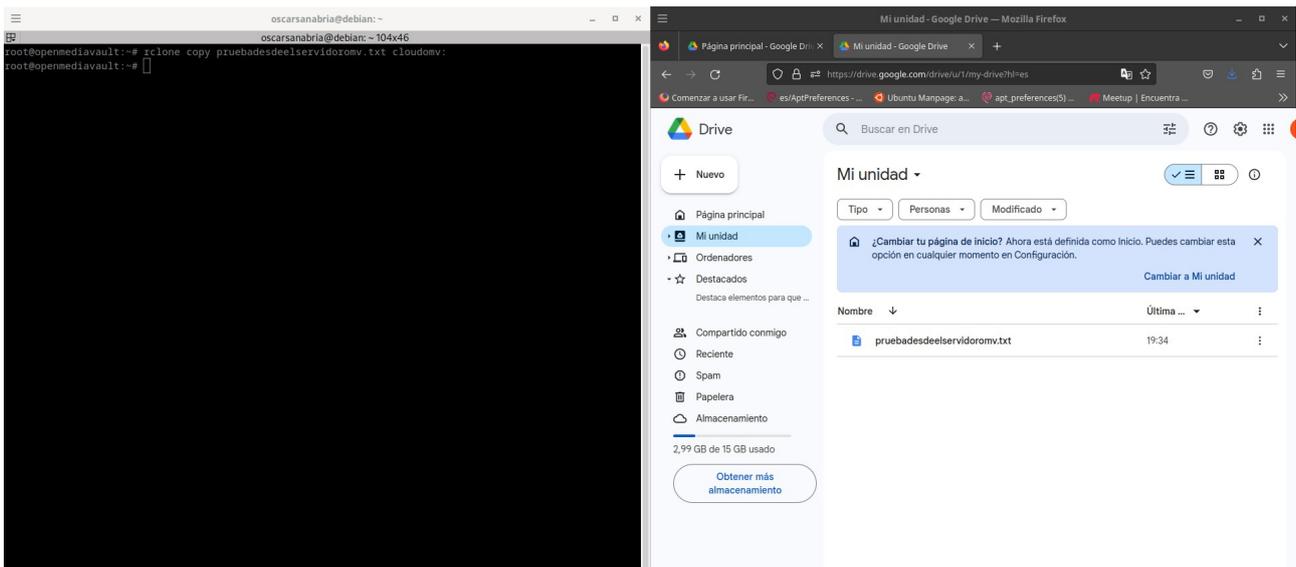
Otorgamos los permisos necesarios a rclone para que pueda acceder y gestionar nuestros archivos en Google Drive.



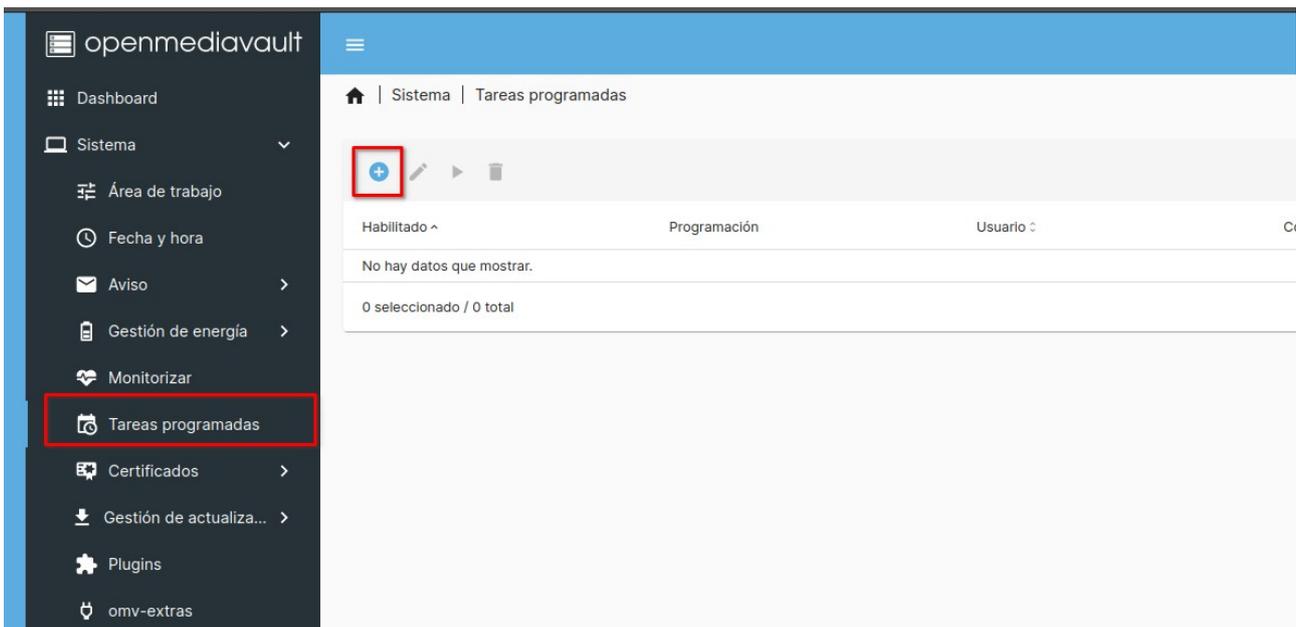
Y podemos confirmar que hemos configurado rclone con éxito.



En tu propia máquina, donde has configurado los permisos de rclone y has otorgado acceso a tu cuenta de Google Drive, recibirás un token. Debes copiar este token y pegarlo en el servidor OMV. Sin este paso, la configuración no se completará correctamente y el servicio no funcionará adecuadamente.



Una vez que tenemos configurado rclone con Google Driver, vamos a configurar una tarea para que realice las copias de seguridad automáticamente en nuestro Google Driver. En el servidor, nos dirigimos al apartado de tareas programadas y creamos una nueva tarea programada.



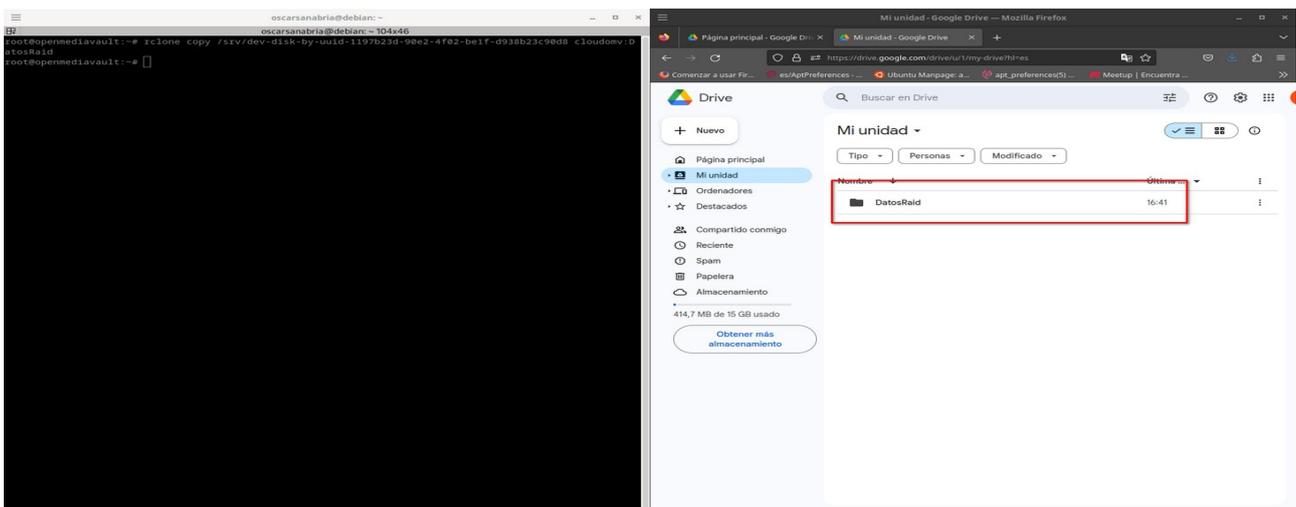
Vamos a configurar una tarea para que realice las copias de seguridad a diario y por ejemplo esa copias de seguridad sea las carpetas compartidas con los clientes Linux y Windows. Pero también podemos realizar copias tanto de configuraciones que tengamos en nuestro servidor omv o datos que sean críticos y necesitemos tener copias de estos datos. Para ello necesitamos el siguiente comando :

- rclone copy

Con este comando haremos la copia de seguridad en nuestro Google driver.

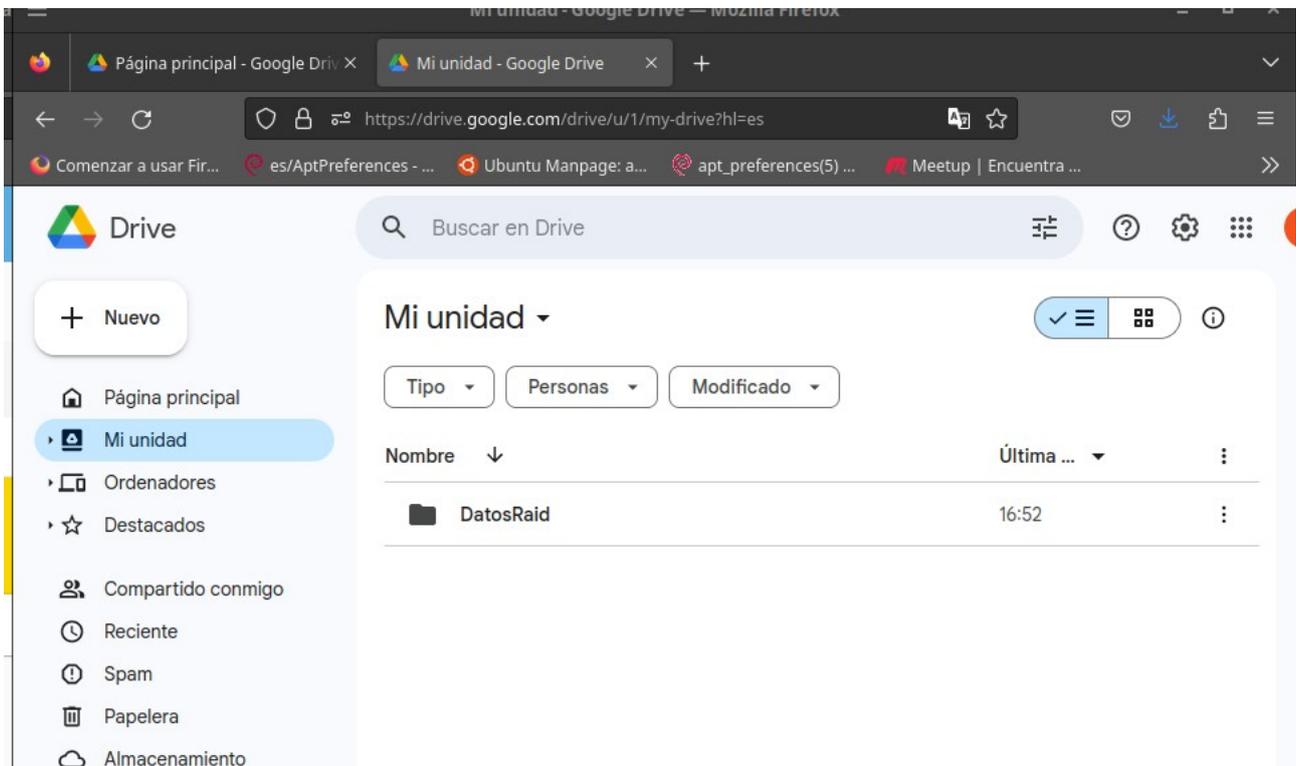
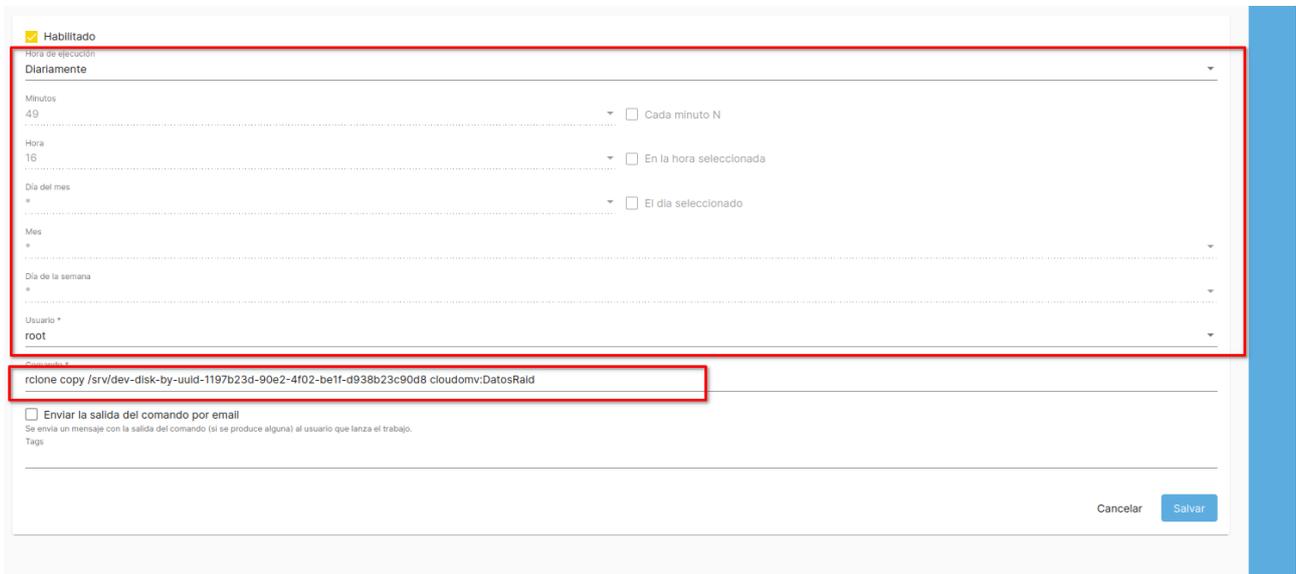
- rclone copy /srv/dev-disk-by-uuid-1197b23d-90e2-4f02-be1f-d938b23c90d8 cloudomv:DatosRaid

He copiado la dirección del raid que tengo montado y donde tenemos las carpetas compartidas para los clientes /srv/dev-disk-by-uuid-1197b23d-90e2-4f02-be1f-d938b23c90d8 después mencionamos el remoto que configuramos anteriormente esto permite la conexión entre Google driver y nuestro servidor **cloudomv** y por ultimo **DatosRaid** , se creara una carpeta en Google driver que se llame **DatosRaid** y tendrá nuestro datos dentro de ella.



Con el comando que hemos utilizamos anteriormente vamos a configurar esta tarea. Se ejecutara todo los días de la semanas y mes a las 16:49.

- **rclone copy /srv/dev-disk-by-uuid-1197b23d-90e2-4f02-be1f-d938b23c90d8 cloudomv:DatosRaid**



Como podemos ver hemos programado una copia de seguridad con omv muy sencillo y más seguro.

Comandos útiles para la gestión con rclone:

Listar ficheros/directorios.

- `rclone ls <proveedor>:<ruta>`

Obtener información de las cuotas de un proveedor

- `rclone about <proveedor>:`

Eliminar un directorio.

- `rclone purge <proveedor>:<ruta>`

Eliminar un Ficheros.

- `rclone delete <proveedor>:<ruta>`

Copiar stdin a un fichero remoto.

- `Lsblk -f | rclone rcat <proveedor>:<ruta>`

Sincronizar un directorio local con un directorio de un proveedores.

- `rclone sync <directorio> <proveedor>:<ruta>`

Montar un directorio remoto en tu ordenador local.

- `rclone mount <proveedor>:<ruta> <montaje> &`

5. Conclusiones y propuestas para seguir trabajando sobre el tema.

A continuación, presentaremos las conclusiones relacionadas con el proyecto:

En primer lugar, la realización del proyecto ha sido positiva. OpenMediaVault (OMV) se ha demostrado como una herramienta muy útil para pequeñas empresas que están comenzando y no disponen de grandes recursos económicos para opciones más costosas. Hemos comprobado que su capacidad para compartir carpetas e incluso discos duros a través de nuestra red nos ofrece grandes ventajas en términos de eficacia en nuestro trabajo y nos permite gestionar nuestro servidor OMV según nuestras necesidades. También hemos observado que se pueden realizar copias de seguridad externas al servidor, lo cual es muy beneficioso, ya que nos brinda la opción de replicar los datos en otro lugar y evitar su pérdida.

En cuanto a los aspectos negativos, es importante mencionar que el servidor requiere mantenimiento. Por lo tanto, el usuario que gestione este servidor debe tener al menos un

conocimiento básico sobre su configuración y administración. Sin embargo, se agradece la disponibilidad de una interfaz web para facilitar estas tareas.

Desde mi punto de vista, OpenMediaVault es una de las opciones más atractivas para el mercado de pequeños servidores. Se recomienda realizar investigaciones más profundas, como la configuración de un orquestador de contenedores que permita interactuar con Kubernetes y tener aplicaciones locales en funcionamiento en nuestro servidor, así como la implementación de conexiones VPN.

6. Dificultades que se han encontrado.

Las dificultades que hemos encontrado en el camino incluyen la falta de documentación detallada para guiar a los usuarios de OpenMediaVault (OMV). Es desconcertante que, a pesar de ofrecer una interfaz web amigable para usuarios con pocos conocimientos técnicos, OMV no proporcione suficiente documentación. La información más comúnmente disponible abarca aspectos básicos como la instalación, el compartir carpetas y la creación de usuarios. Sin embargo, si se desea ir más allá de estos temas, es muy difícil encontrar información adecuada, y la que se encuentra suele estar mal explicada.

La configuración de los plugins, que son fundamentales para darle vida y funcionalidad a OMV y por lo que esta plataforma destaca, carece de guías claras. La falta de explicaciones accesibles hace que los usuarios con menos experiencia tengan dificultades para comprender y manejar estos aspectos más complejos.

7. Bibliografía, enlaces, reseñas

- [OpenMediaVault - Wikipedia](#)
- [OpenMediaVault Documentation: Releases](#)
- [OpenMediaVault Documentation: General Administration Workbench](#)
- [Storage Area Network \(SAN\) vs. Network Attached Storage \(NAS\) - FS Community](#)
- [Sistema de Archivos - Universidad de Murcia](#)
- [OpenMediaVault Documentation: iSCSI Configuration](#)
- [Docker Plugin for OpenMediaVault](#)
- [Rclone for Backup in OpenMediaVault](#)
- [Guía oficial omv-extras](#)
- [OpenMediaVault Documentation: RAID Configuration](#)
- [OpenMediaVault Documentation: LVM Configuration](#)
- [OpenMediaVault Documentation: File System Management](#)
- [OpenMediaVault Documentation: User Management](#)