

Protocolos de transferencia de archivos:

FTP, SFTP y SSH



DAVID SEGURA TRISTANCHO
2º ASIR 2022/23

ÍNDICE

1. ¿Qué es FTP?	2
2. Servidor FTP y Filezilla	2
3. ¿Qué es Open Media Vault?	5
4. Open Media Vault	5
• RAID	5
• Sistema de archivos	6
• Carpetas compartidas	7
• Servicios FTP	7
• Permisos, Usuarios y Grupos	8
• Certificados	10
• Otros aspectos	11
5. ¿Qué es SSH?	12
6. SSH en Ubuntu Server	12
⑩ Conexión sin tener que hacer uso de contraseña:	13
⑩ Conexión sin tener que hacer uso de contraseña mediante scp:	14

1. ¿Qué es FTP?

FTP (File Transfer Protocol o Protocolo de Transferencia de Archivos) es un protocolo de red que empezó a utilizarse en el año 1974, su principal objetivo, es la transferencia de archivos entre dos equipos.

El servicio FTP es ofrecido por la capa de Aplicación del modelo TCP/IP y utiliza normalmente el puerto 20 y el 21 para conectarse. Hoy en día, es muy habitual emplear gestores de FTP como por ejemplo FileZilla o WinSCP, para subir los contenidos al hosting o administrar sus ficheros de forma segura.

2. Servidor FTP y Filezilla

Abro Ubuntu Server y me conecto desde la terminal principal mediante SSH. Comienzo por algo siempre útil como es actualizar los repositorios e instalo el servicio de FTP.

```
usuario@usuario:~$ sudo apt install vsftpd
```

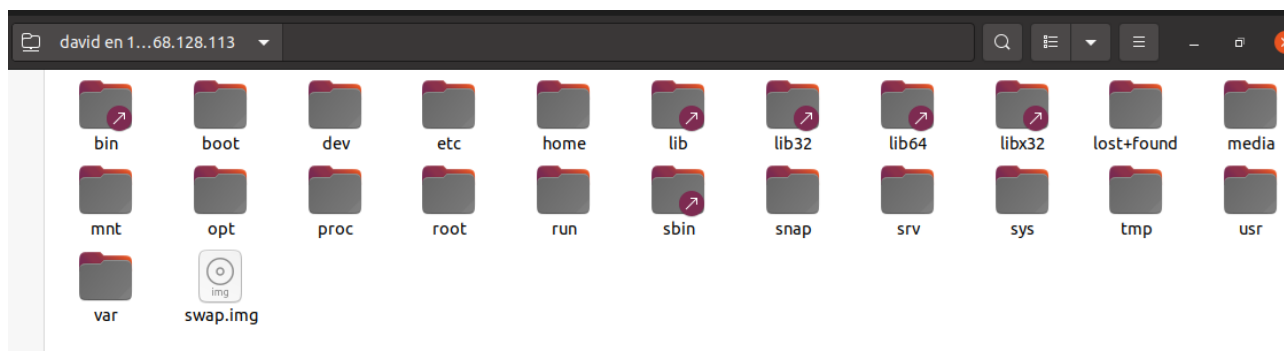
Comprobamos los puertos existentes para saber el de FTP.

```
usuario@usuario:~$ sudo netstat -natup
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 10.10.10.3:53          0.0.0.0:*               LISTEN      631/named
tcp        0      0 192.168.128.113:53     0.0.0.0:*               LISTEN      631/named
tcp        0      0 127.0.0.1:53          0.0.0.0:*               LISTEN      631/named
tcp        0      0 127.0.0.53:53         0.0.0.0:*               LISTEN      604/systemd-resolve
tcp        0      0 0.0.0.0:22            0.0.0.0:*               LISTEN      709/sshd: /usr/sbin
tcp        0      0 127.0.0.1:953         0.0.0.0:*               LISTEN      631/named
tcp        0      36 192.168.128.113:22     192.168.128.14:54666    ESTABLISHED 1080/sshd: usuario
tcp6       0      0 :::80                 :::*                   LISTEN      742/apache2
tcp6       0      0 :::21                 :::*                   LISTEN      1764/vsftpd
tcp6       0      0 fe80::a00:27ff:feec::53 :::*                   LISTEN      631/named
tcp6       0      0 fe80::a00:27ff:feca::53 :::*                   LISTEN      631/named
tcp6       0      0 :::1:53               :::*                   LISTEN      631/named
tcp6       0      0 :::22                 :::*                   LISTEN      709/sshd: /usr/sbin
tcp6       0      0 :::1:953              :::*                   LISTEN      631/named
udp        0      0 10.10.10.3:53          0.0.0.0:*               631/named
udp        0      0 192.168.128.113:53     0.0.0.0:*               631/named
udp        0      0 127.0.0.1:53          0.0.0.0:*               631/named
udp        0      0 127.0.0.53:53         0.0.0.0:*               604/systemd-resolve
udp        0      0 192.168.128.113:68     0.0.0.0:*               602/systemd-network
udp6       0      0 :::1:53               :::*                   631/named
udp6       0      0 fe80::a00:27ff:feca::53 :::*                   631/named
udp6       0      0 fe80::a00:27ff:feec::53 :::*                   631/named
```

Creo un nuevo usuario con el que poder realizar diferentes probaturas:

```
usuario@usuario:/$ sudo adduser david
Adding user `david' ...
Adding new group `david' (1001) ...
Adding new user `david' (1001) with group `david' ...
The home directory `/home/david' already exists. Not copying from `/etc/skel'.
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for david
Enter the new value, or press ENTER for the default
  Full Name []:
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] Y
```

Accedo a Otras ubicaciones, añado ftp:// seguido de la ip del servidor. Accedo como usuario registrado.

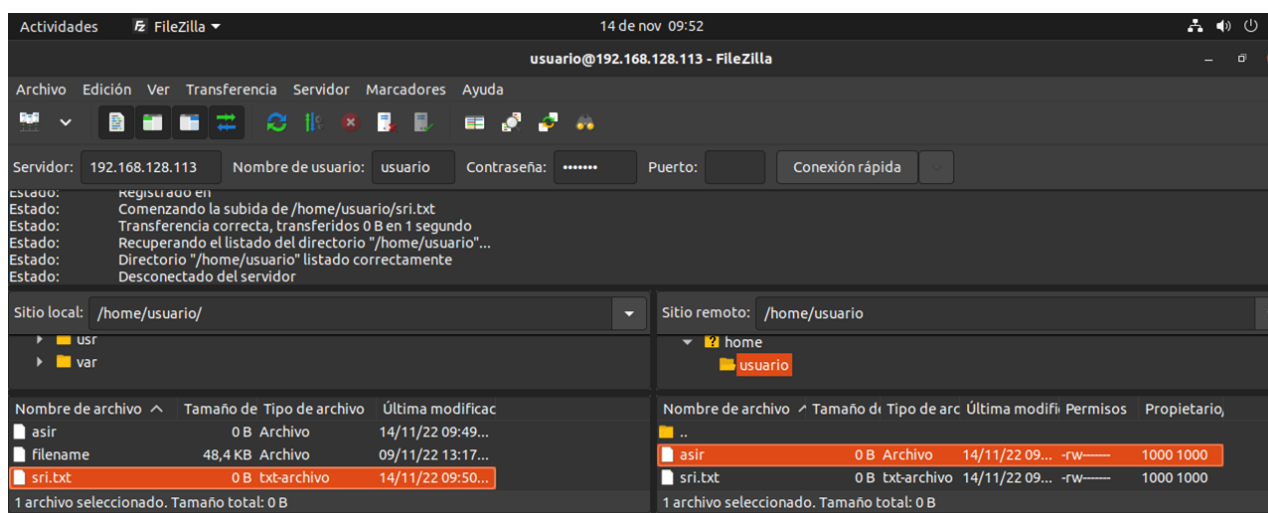


Ahora abro un Ubuntu Desktop como cliente, de nuevo actualizo repositorios e instalo el programa Filezilla. FileZilla es un software gratuito que funciona a nivel cliente/servidor. Significa que nosotros, como usuarios, podemos conectarnos a un servidor para consultar, adquirir y manipular contenido del mismo. Resulta realmente útil si queremos tener una conexión directa a una fuente de datos sin perder excesivo tiempo en el proceso.

```
usuario@aso:~$ sudo apt install filezilla
```

Acto seguido, abro Filezilla y dentro del entorno gráfico añado el servidor, usuario, contraseña de usuario y el puerto (21), si fuera necesario. Luego me conecto, y si creo un archivo desde la terminal del cliente y refresco Filezilla me aparecerá.

```
usuario@aso:~$ sudo touch sri.txt
```



En la esquina inferior izquierda de la última imagen, podemos ver como el archivo sri.txt creado desde la terminal ha aparecido automáticamente en Filezilla justo después de actualizarlo.

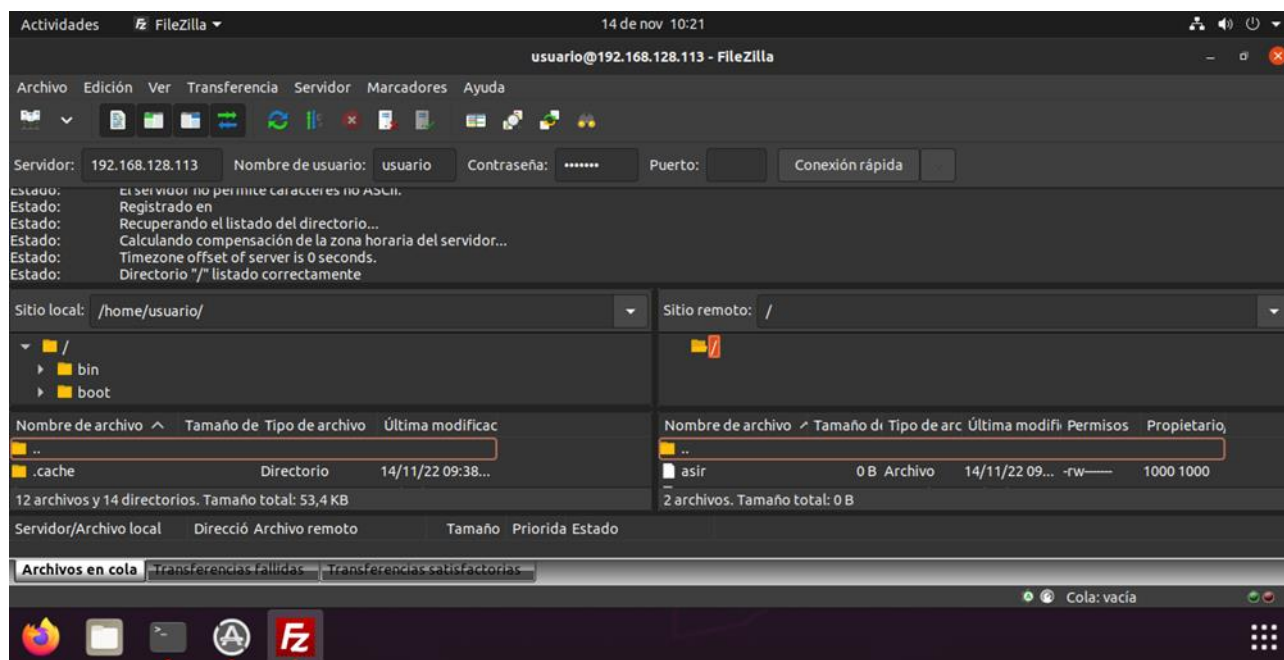
A continuación, accedo al servidor y me voy al directorio /etc. Dentro de /etc voy al archivo de configuración de vsftpd. En la zona de chroot, descomento la línea de chroot local user y la pongo en YES. Igualmente, añado una segunda línea, allow_writeable_chroot=YES.

```
# You may fully customize the login banner string.
#ftpd_banner=Welcome to blah FTP service.
#
# You may specify a file of disallowed anonymous e-mail addresses. Apparently
# useful for combatting certain DoS attacks.
#deny_email_enable=YES
# (default follows)
#banned_email_file=/etc/vsftpd.banned_emails
#
# You may restrict local users to their home directories. See the FAQ for
# the possible risks in this before using chroot_local_user or
# chroot_list_enable below.
chroot_local_user=YES
allow_writeable_chroot=YES
#
# You may specify an explicit list of local users to chroot() to their home
# directory. If chroot_local_user is YES, then this list becomes a list of
# users to NOT chroot().
# (Warning! chroot'ing can be very dangerous. If using chroot, make sure that
# the user does not have write access to the top level directory within the
# chroot)
```

Reinicio el servicio y lo compruebo:

```
usuario@usuario:/etc$ sudo service vsftpd restart
usuario@usuario:/etc$ sudo service vsftpd status
```

Con esto, lo que he hecho es indicar que el usuario solo puede acceder a los directorios de su carpeta home. Si refresco el Filezilla o cierro y vuelvo a abrirlo, veremos cómo se produce:



Visto este último detalle, dejamos FTP y Filezilla para dar paso a OMV.

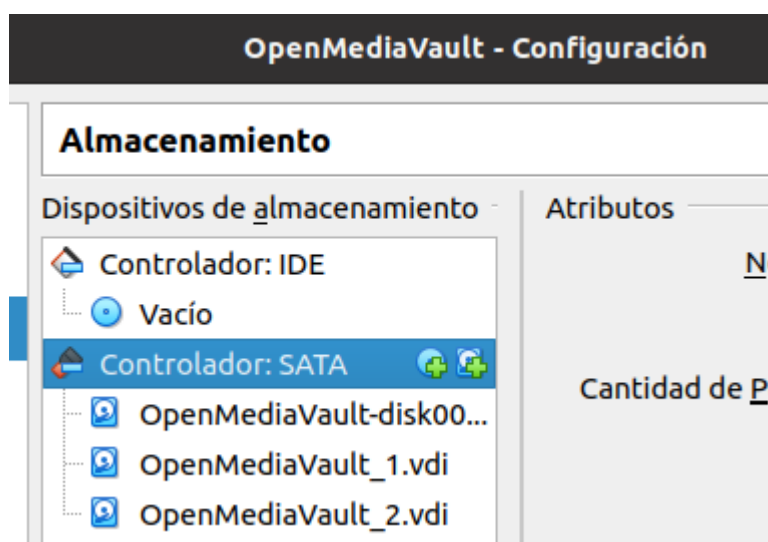
3. ¿Qué es Open Media Vault?

Si nunca has utilizado OMV, debes saber que es un sistema operativo con una interfaz gráfica muy intuitiva que nos permitirá tener un completo servidor NAS con una gran cantidad de servicios. Podremos configurar en detalle la red, los servicios de red que queramos servir (Samba, FTP, FTPES, SSH, SFTP, Rsync, NFS v3 y NFSv4 entre otros muchos), también podremos monitorizar en detalle el estado del sistema, tanto a nivel de los discos duros o SSD, como del sistema completo.

Otras características interesantes son que permite utilizar SNMP para monitorizarlo externamente, e incluso permite configurar notificaciones por email.

4. Open Media Vault

Abrimos nuestra maquina OMV y la ip la colocamos en el navegador. Accedemos a la interfaz gráfica y volvemos a salir. La máquina se cerrará automáticamente. Nos vamos a almacenamiento, y en SATA añado dos discos más.

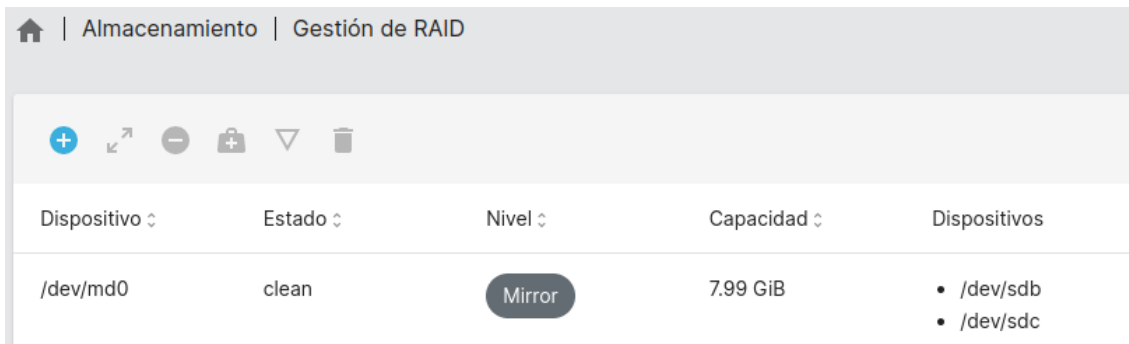


- **RAID**

Vamos a almacenamiento, gestión RAID, creamos un raid espejo y añadimos los dos discos creados:

Almacenamiento Discos				
Dispositivo ^	Modelo ^	Número de Serie ^	Vendedor ^	Capacidad ^
/dev/sda	VBOX HARDDISK	VB752841be-6be43b32	ATA	8.00 GiB
/dev/sdb	VBOX HARDDISK	VB58a953be-cd7b1937	ATA	8.00 GiB
/dev/sdc	VBOX HARDDISK	VB9f7c6708-1000f7dc	ATA	8.00 GiB
0 Seleccionado / 3 total				

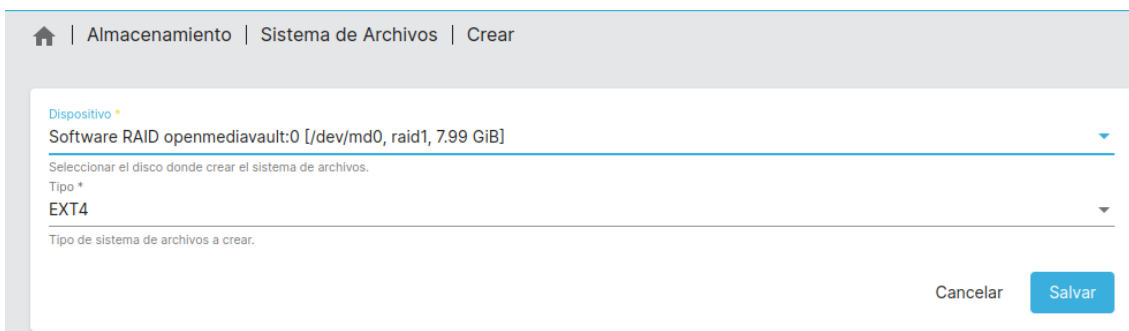
Salvamos y aplicamos. Importante guardar y después aplicar cada proceso efectuado:



Dispositivo	Estado	Nivel	Capacidad	Dispositivos
/dev/md0	clean	Mirror	7.99 GiB	<ul style="list-style-type: none"> /dev/sdb /dev/sdc

• Sistema de archivos

Vamos a Sistema de archivos y lo creamos para md0:



Almacenamiento | Sistema de Archivos | Crear

Dispositivo
Software RAID openmediavault:0 [/dev/md0, raid1, 7.99 GiB]

Selecciónar el disco donde crear el sistema de archivos.

Tipo
EXT4

Tipo de sistema de archivos a crear.

Cancelar Salvar

Salvamos de nuevo y aplicamos los cambios:

Crear un sistema de archivos

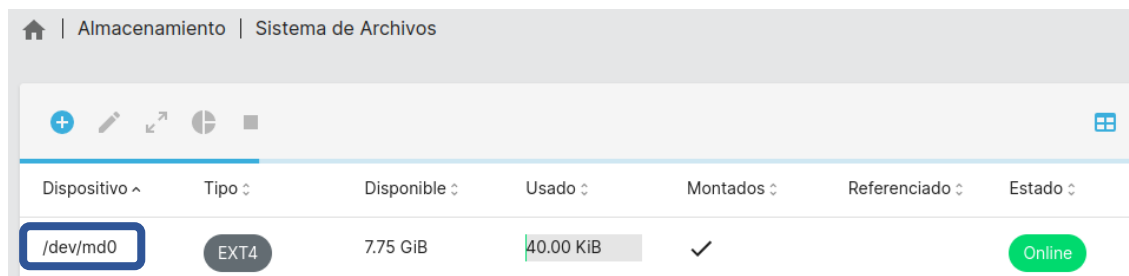
```
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 2094848 4k blocks and 524288 inodes
Filesystem UUID: b73e7af3-0d79-4361-afe6-905550951ad2
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632

Allocating group tables: 0/64          done
Writing inode tables: 0/64             done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: 0/64      done

END OF LINE
```

Y como se puede apreciar, ya tenemos montado nuestro sistema de archivos:

Cerrar



Dispositivo	Tipo	Disponible	Usado	Montados	Referenciado	Estado
/dev/md0	EXT4	7.75 GiB	40.00 KiB	✓		Online

• Carpetas compartidas

Vamos a carpetas compartidas y creamos una carpeta llamada DATOS a la cual le vamos a añadir el sistema de archivos creado anteriormente. También especificamos en caso de ser requerido los permisos pertinentes:

Almacenamiento | Carpetas Compartidas | Crear

Nombre *
DATOS

Sistema de Archivos *
/dev/md0 [EXT4, 40.00 KiB (1%) used, 7.75 GiB available]

El sistema de archivos en el que será creada la carpeta compartida.
ruta relativa *
DATOS/

La ruta relativa de la carpeta para compartir. La carpeta especificada se creará si no existe.
Permisos *
Administrador: Lectura/Escritura, Usuarios: Lectura/Escritura, Otros: Solo lectura

Modo de archivo de la ruta a las carpetas compartidas.
Comentario

Indicamos los parámetros y salvamos:

Almacenamiento | Carpetas Compartidas

Nombre ^	Dispositivo ↕	Ruta relativa ↕	Ruta absoluta ↕	Referenciado ↕	Comentario ↕
DATOS	/dev/md0	DATOS/	/srv/dev-disk-by-uuid-b73e7af3-0d79-4361-afe6-905550951ad2/DATOS		

0 Seleccionado / 1 total

Si en todas las ubicaciones inserto en el recuadro de búsqueda ftp:// + la ip de mi OMV puedo acceder a mi carpeta recién creada llamada DATOS. De este modo verificamos que se ha creado y que tengo permitido el acceso.

• Servicios FTP

Vamos a servicios, FTP, configuración y habilitamos el servicio:

Servicios | FTP | Configuración

☒ **Habilitado**

Port *
21

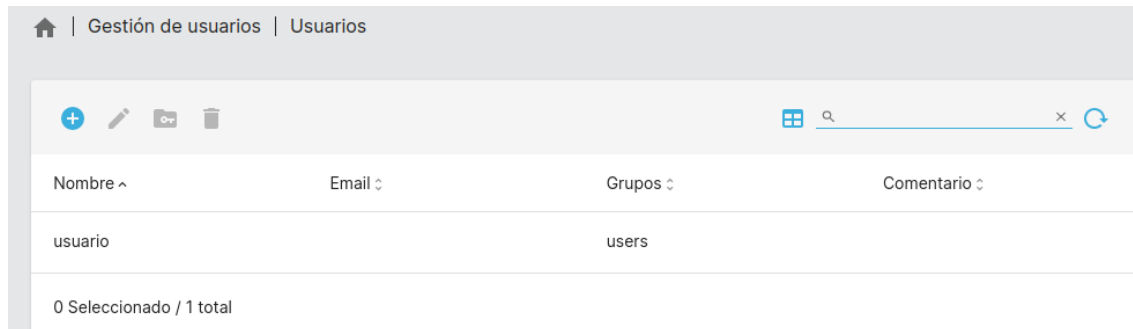
Max. clients *
5
Maximum number of simultaneous clients.

Max. connections per host *
2
Maximum number of connections per IP (0 = unlimited).

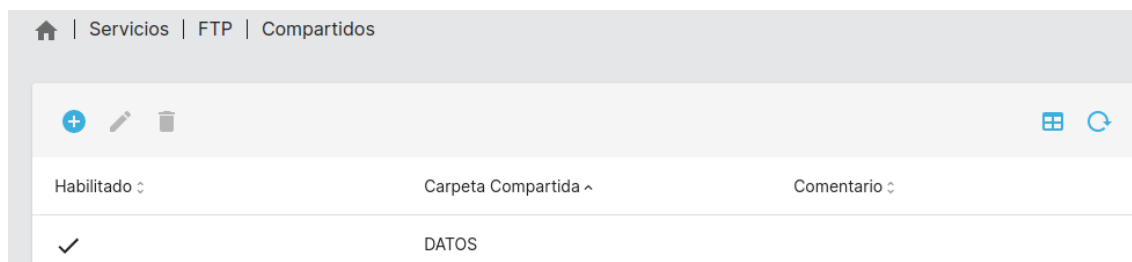
Max. login attempts *
1
Maximum number of allowed password attempts before disconnection.

Timeout *
1200
Maximum idle time in seconds. Setting idle timeout to 0 disables the idle timer completely (clients can stay connected for ever, without sending data).

Creo un usuario en Gestión de usuarios:

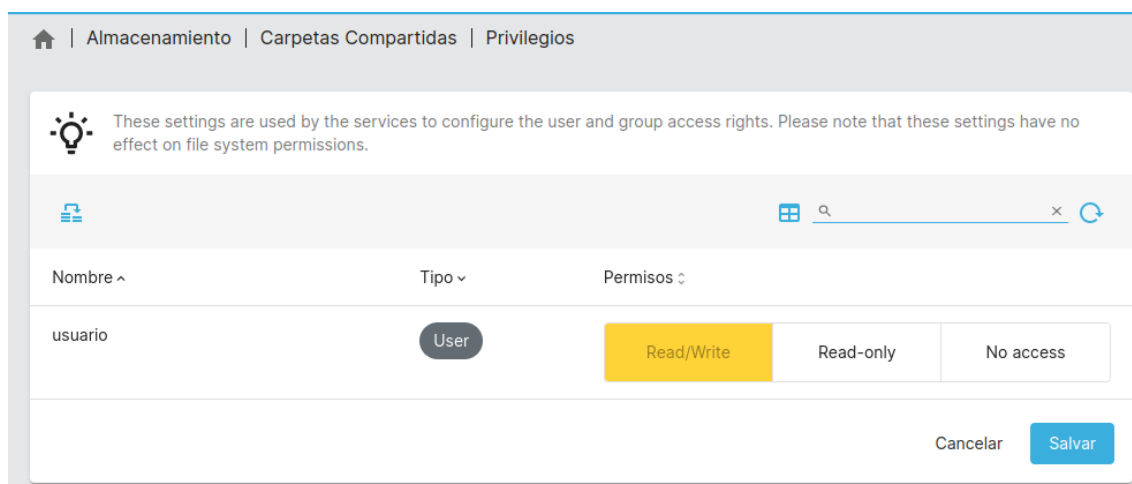


De ahí pasamos a Servicios, FTP y compartidos. Ya aparece la carpeta compartida previamente (DATOS):



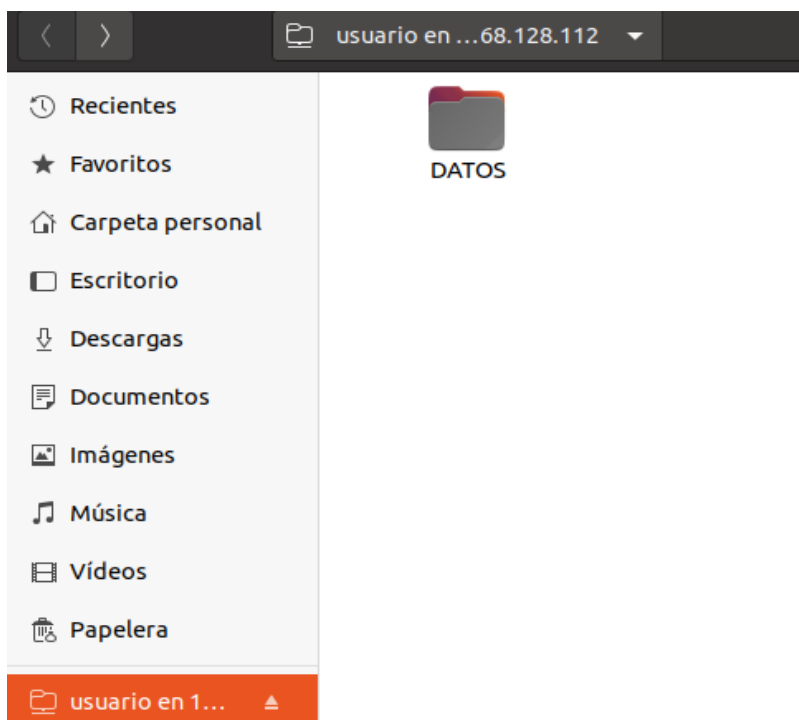
- **Permisos, Usuarios y Grupos**

En almacenamiento, carpetas compartidas, privilegios, damos al usuario permisos de escritura y lectura:

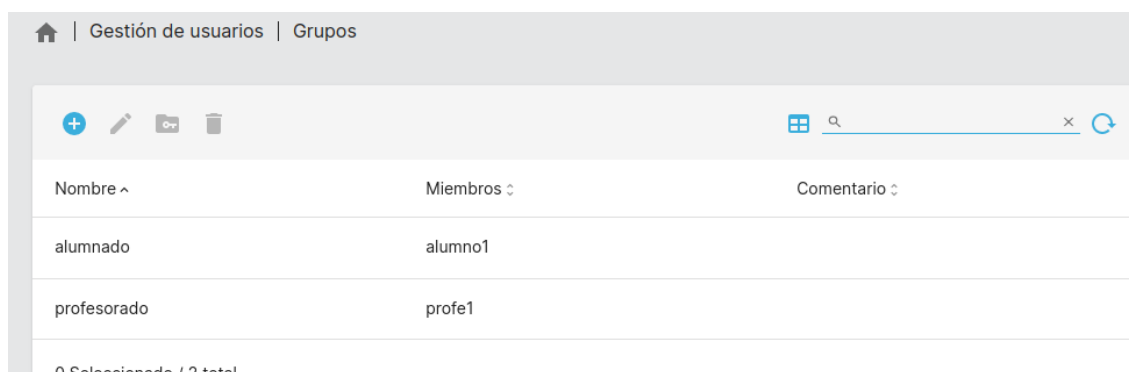


Como queda reflejado según la imagen, he concedido permisos de lectura y escritura al usuario en cuestión.

De hecho, si accedo al nas con usuario y su ip me cerciorare de que puedo acceder a la carpeta compartida DATOS e incluso visualizar el contenido de sus documentos:



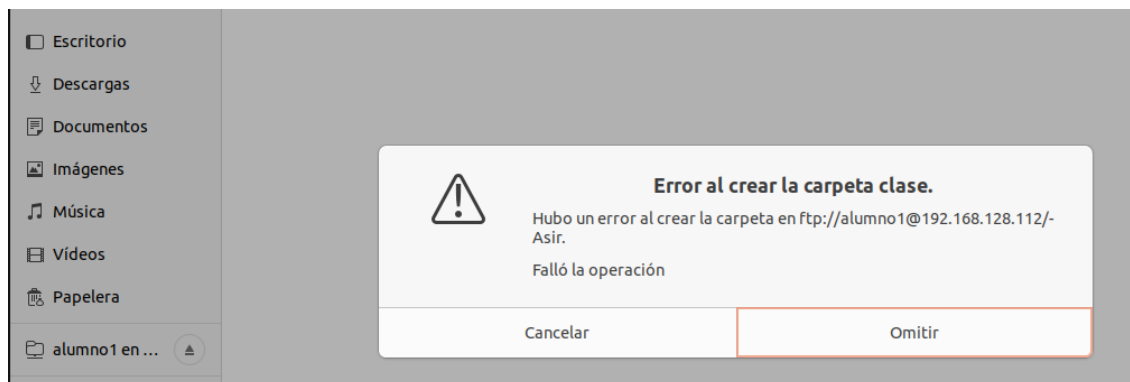
Seguimos viendo las posibilidades que ofrece OMV y vamos a proceder a crear dos grupos, uno para profesorado y otro para alumnado:



Creo otra carpeta compartida llamada Asir:



Si accedo mediante nas al alumno 1, comprobaré como no puedo crear una carpeta dentro de Asir al no tener los permisos para ello:



• Certificados

A continuación, creo un certificado para FTP en SLL y lo habilito:



Si accedo a Filezilla y establezco conexión me pedirá aceptarla en base al certificado creado:



- Otros aspectos

En servicios, FTP, también podemos configurar aspectos como el FTP pasivo. Para que nos entendamos, el servidor FTP puede ser Activo o Pasivo, o ambos. En una conexión FTP activa, el cliente abre un puerto y escucha y el servidor se conecta activamente a él. En una conexión FTP pasiva, el servidor abre un puerto y escucha (pasivamente) y el cliente se conecta a él.

The screenshot shows the 'Configuración' (Configuration) page for FTP services. The breadcrumb trail is 'Servicios | FTP | Configuración'. The 'FTP Pasivo' section is active, indicated by a checked checkbox. Below the checkbox, there is explanatory text in Spanish about passive ports and firewall limitations. The configuration fields are as follows:

Field	Value
Min. passive port *	49152
Max. passive port *	65534
Dirección IP enmascarada	
Refresh time	0

En este apartado, podemos configurar cuestiones como el mínimo y máximo de puertos pasivos, así como la dirección IP enmascarada y el tiempo de refresco. En este caso, por ejemplo, lo he habilitado y lo he dejado tal y como venía por defecto.

E igualmente, podemos añadir ciertas reglas de baneo. Por ejemplo, yo he estipulado que solo podrá haber un máximo de dos conexiones por usuario:

The screenshot shows the 'Crear' (Create) page for ban rules. The breadcrumb trail is 'Servicios | FTP | Reglas de baneo | Crear'. The rule is named 'MaxHostsPerUser'. The configuration details are as follows:

Field	Value
Evento	MaxHostsPerUser
Esta regla se disparará en cuanto ocurra el evento seleccionado.	
Occurrence *	2
This parameter says that if N occurrences of the event happen within the given	
Intervalo *	00:30:00
Especifica el tiempo en hh:mm:ss en el que debe ocurrir la incidencia para bane	
Caduca *	00:10:00
Especifica el tiempo en hh:mm:ss para que expire el baneo	

5. ¿Qué es SSH?

SSH o Secure Shell, es un protocolo de administración remota que le permite a los usuarios controlar y modificar sus servidores remotos a través de Internet a través de un mecanismo de autenticación. Proporciona un mecanismo para autenticar un usuario remoto, transferir entradas desde el cliente al host y retransmitir la salida de vuelta al cliente.

El servicio se creó como un reemplazo seguro para el Telnet sin cifrar y utiliza técnicas criptográficas para garantizar que todas las comunicaciones hacia y desde el servidor remoto sucedan de manera encriptada.

6. SSH en Ubuntu Server

Instalamos el servicio, con **sudo apt install ssh**, comprobamos que el servicio está activo y nos vamos a `/etc/ssh`. Si hago un listado, podemos ver lo siguiente:

```
usuario@usuario:/etc/ssh$ ls -l
total 576
-rw-r--r-- 1 root root 535195 mar 30 2022 moduli
-rw-r--r-- 1 root root 1603 mar 30 2022 ssh_config
drwxr-xr-x 2 root root 4096 mar 30 2022 ssh_config.d
-rw-r--r-- 1 root root 3289 mar 30 2022 sshd_config
drwxr-xr-x 2 root root 4096 mar 30 2022 sshd_config.d
```

Los archivos `ssh_config` y `sshd_config` pertenecen a servidor y cliente.

A continuación, hago una comprobación de los puertos existentes. Como puede verse, tengo el puerto ssh (22) abierto.

```
usuario@usuario:/etc/ssh$ sudo netstat -natup
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53          0.0.0.0:*               LISTEN      606/systemd-resolve
tcp        0      0 0.0.0.0:22            0.0.0.0:*               LISTEN      701/sshd: /usr/sbin
tcp        0      0 192.168.128.171:22     192.168.128.14:45814   ESTABLISHED 1260/sshd: usuario
tcp        0      0 192.168.128.171:58390  185.125.190.39:80     TIME_WAIT   -
tcp6       0      0 :::22                 :::*                   LISTEN      701/sshd: /usr/sbin
tcp6       0      0 :::80                 :::*                   LISTEN      717/apache2
udp        0      0 127.0.0.53:53          0.0.0.0:*               606/systemd-resolve
```

Si me voy a la máquina anfitriona de alumnado y llevo a cabo la siguiente secuencia de comandos llegaré al archivo de known hosts.

```
alumnado@A128PC04:~$ cd .ssh/
alumnado@A128PC04:~/.ssh$ ls -la
total 12
drwx----- 2 alumnado alumnado 4096 nov 21 09:18 .
drwxr-xr-x 29 alumnado alumnado 4096 nov 18 10:50 ..
-rw-r--r-- 1 alumnado alumnado 2664 oct 24 09:05 known_hosts
```

Aquí se encuentran las claves para la conexión ssh a otra máquina en base a pedir la contraseña.

En otro orden de factores, puedo cambiar el puerto descomentandolo y aplicando el siguiente comando:

```
usuario@usuario:/etc/ssh$ sudo nano sshd_config
usuario@usuario:/etc/ssh$ sudo service ssh restart
usuario@usuario:/etc/ssh$ ssh -p 2222 usuario@192.168.128.171
```

Reinicio el servicio y para permitir hacer conexión ssh con root accedo al archivo sshd_config:

```
GNU nano 4.8 sshd_config
# default value.

Include /etc/ssh/sshd_config.d/*.conf

Port 2222
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:
#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10
```

Esto se ha hecho en clase a modo de prueba, posteriormente he vuelto a restaurar los cambios dejándolos por defecto.

⑩ Conexión sin tener que hacer uso de contraseña:

En primer lugar, generamos la clave pública con el siguiente comando:

```
alumnado@A128PC04:~$ ssh-keygen -b 4096 -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/alumnado/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/alumnado/.ssh/id_rsa
Your public key has been saved in /home/alumnado/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:X7SjvHDhGLIQ6wM4387r0m+jIz9WcsJu/cQGek7zj0Y alumnado@A128PC04
The key's randomart image is:
+----[RSA 4096]-----+
|
|  .      .
| .  o    . .
|o.. + . S . +
| o+=o+ o * + .
| .o*B = o *
| . B*=0 o .
| =+%E++ .
+----[SHA256]-----+
```

Acto seguido la copiamos a la otra máquina con el siguiente comando:

```
- ssh-copy-id usuario@192.168.128.171
```

Y nos aparecerá en la máquina de usuario el archivo known_hosts.

```
usuario@usuario:~$ cd .ssh/  
usuario@usuario:~/.ssh$ ls -la  
total 16  
drwx----- 2 usuario usuario 4096 nov 21 08:46 .  
drwxr-xr-x 4 usuario usuario 4096 oct 10 11:06 ..  
-rw----- 1 usuario usuario 743 nov 21 08:55 authorized_keys  
-rw-r--r-- 1 usuario usuario 222 nov 21 08:26 known_hosts
```

Si accedo a su interior, podré ver la clave creada.

```
GNU nano 4.8      known_hosts  
1|wkfgY/kf1XBe9WK0rJVuXr9MU80=|2278e042s4EYc1uaCmoLJdronq4= ecdsa-sha
```

Y ahora sí, puedo acceder via ssh desde alumnado a mi máquina de usuario:

```
alumnado@A128PC04:~$ ssh usuario@192.168.128.171  
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.4.0-126-generic x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:       https://ubuntu.com/advantage
```

⑩ Conexión sin tener que hacer uso de contraseña mediante scp:

Hago un listado en mi máquina de alumnado y hago el scp con el archivo id_rsa.pub.

```
alumnado@A128PC04:~/.ssh$ ls  
id_rsa id_rsa.pub known_hosts  
alumnado@A128PC04:~/.ssh$ scp id_rsa.pub usuario@192.168.128.171:  
id_rsa.pub                                100% 743 375.3KB/s 00:00
```

Si me voy al usuario, al servidor, en .ssh tengo ya copiado el archivo id_rsa.pub

```
usuario@usuario:~/.ssh$ cd  
usuario@usuario:~$ ls -la  
total 44  
drwxr-xr-x 5 usuario usuario 4096 nov 21 09:18 .  
drwxr-xr-x 11 root root 4096 nov 17 12:27 ..  
-rw----- 1 usuario usuario 1406 nov 21 09:15 .bash_history  
-rw-r--r-- 1 usuario usuario 220 feb 25 2020 .bash_logout  
-rw-r--r-- 1 usuario usuario 3771 feb 25 2020 .bashrc  
drwx----- 2 usuario usuario 4096 sep 30 11:43 .cache  
-rw-r--r-- 1 usuario usuario 743 nov 21 09:18 id_rsa.pub
```


Observo en `authorized_keys` cómo la clave se me ha introducido y autorizado:

```
usuario@usuario: ~/.ssh 70x38
GNU nano 4.8 authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDdAa3qe10HwKMR9cpSanV2IUhoROdbF>
```

Paso la clave `id` al archivo de `authorized keys`:

```
usuario@usuario:~$ cat id_rsa.pub > .ssh/authorized_keys
```

Y después de todo este proceso, podré acceder desde `alumnado` a mi máquina de usuario sin que tenga que introducir contraseña.

```
alumnado@A128PC04:~$ ssh usuario@192.168.128.171
Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.4.0-126-generic x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage
```

