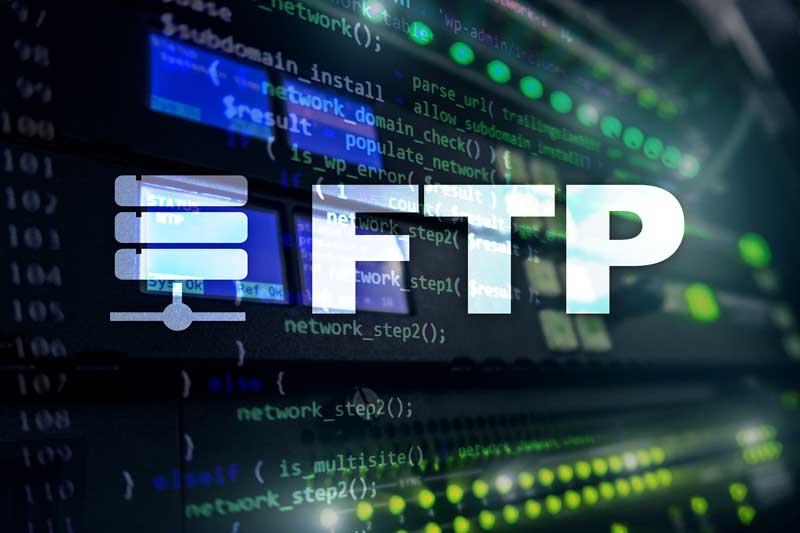
VSFTP



David Fidel Apaza Mamani

## 

[**1º Instalación de vsftpd 3**](#_o38lyzi09xdl)

[**2º Creación de usuarios y grupos 4**](#_y3ynl3308715)

[**3º Configuración de accesos a los directorios 4**](#_8nb83hy6mm6r)

[**4º Configuración de vsftpd.conf 5**](#_c83sgyk828mh)

[FTPS 6](#_f26y31o9dh6c)

[SFTP 8](#_g6dbmeh1jovy)

[**5º Pruebas 9**](#_mc6y3x48tsu7)

[Filezilla 9](#_bcyw2aepbgy6)

[FTP plano 9](#_hsdf9em4lwk0)

[FTPS 15](#_2wwip2ch7z8y)

[CMD 16](#_9fuhw2m1bat6)

[WinSCP 18](#_4fc0bsiqpf12)

[Wireshark 20](#_j9bsjfh5ilu4)

[**Conclusión 24**](#_jp75eyjoymea)

[Aspectos Positivos: 24](#_i31ryd20t8lz)

[1. Seguridad: 24](#_7tf7j63g6pue)

[2. Rendimiento: 24](#_jbd0ss6o5ere)

[3. Protocolo FTPS: 24](#_afmajycqjgca)

[4. Compatibilidad con SFTP: 24](#_guqtx6vh4uxd)

[5. Configuración Personalizada: 24](#_d3n60hlive3p)

[Consideraciones y Mejoras Potenciales: 25](#_guw0o7mj0a1b)

[1. Documentación: 25](#_dduxsqox7kc)

[2. Monitoreo y Registro: 25](#_b66otwqwqr1n)

[3. Gestión de Usuarios: 25](#_k9vhp6l6f8bc)

## 

## 1º Instalación de vsftpd

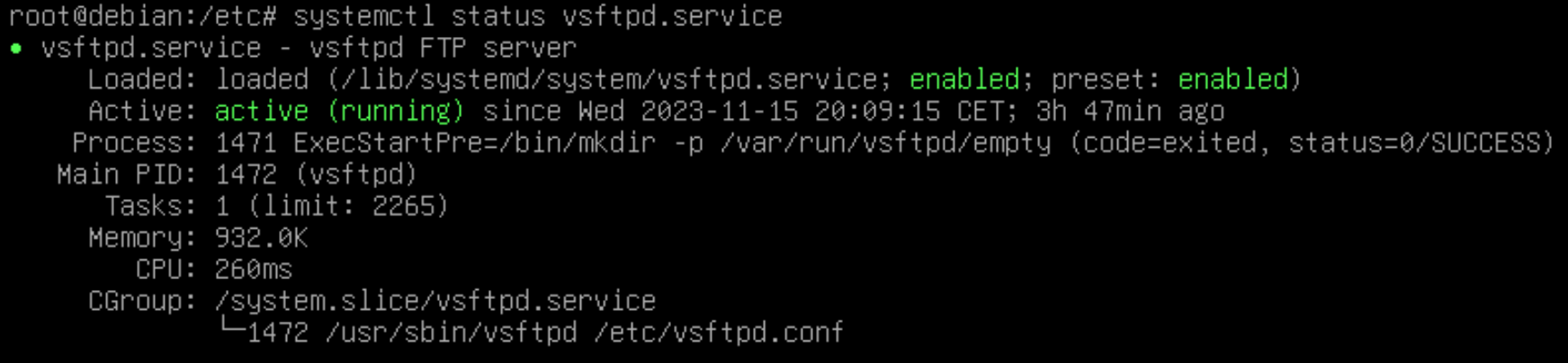
Actualizamos los repositorios

***sudo apt update***

Instalamos vsftpd

***sudo apt install vsftpd***

Comprobamos el estado del servicio



## 

## 

## 2º Creación de usuarios y grupos

Creamos el grupo "alumnos".

***sudo groupadd alumnos***

Creamos el grupo "profesores".

***sudo groupadd profesores***

Creamos al usuario “matio”con su propio directorio y le asignamos una contraseña.

***sudo useradd -m -G alumnos matio***

***sudo passwd matio***

Creamos al usuario “edu” con su propio directorio y le asignamos una contraseña.

***sudo useradd -m -G profesores edu***

***sudo passwd edu***

Añadimos a “edu“ al grupo alumnos (para permitir acceso a directorios de alumnos), ahora “edu” está en tres grupos (edu, profesores y alumnos).

***sudo usermod -aG alumnos edu***



## 3º Configuración de accesos a los directorios

Vamos a cambiar el propietario y el grupo del directorio de “matio” de esta manera “matio” tiene todos los permisos de su directorio y los integrantes del grupo “alumnos” pueden leer pero no escribir.

sudo chown matio:alumnos /home/matio

sudo chmod 750 /home/matio



En el directorio de “edu” cambiamos el propietario y grupo de esta manera “edu” tiene todos los permisos y los miembros del grupo profesores tienen permiso de lectura y ejecución.

sudo chown edu:profesores /home/edu

sudo chmod 750 /home/edu

## 4º Configuración de vsftpd.conf

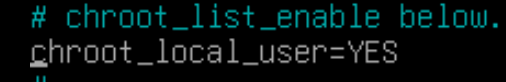
Creamos una copia de respaldo del archivo de configuración.

***sudo cp /etc/vsftpd.conf /etc/vsftpd.conf.original***

Editamos el archivo de configuración ‘/etc/vsftpd.conf’

***sudo nano /etc/vsftpd.conf***

Descomentamos **chroot\_local\_user=YES** para que los usuarios permanezcan restringidos en su propio directorio en /home

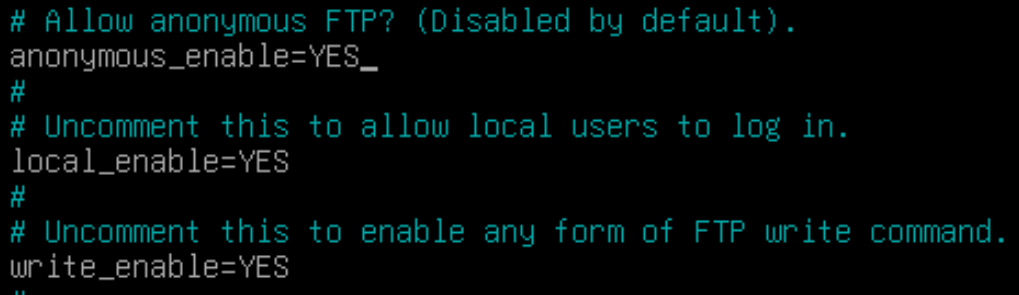


Permitimos el acceso a usuarios locales, anónimos y habilitamos la escritura en el servicio vsftpd aunque no todos los usuarios podrán escribir en todos los directorios.

**anonymous\_enable=YES**

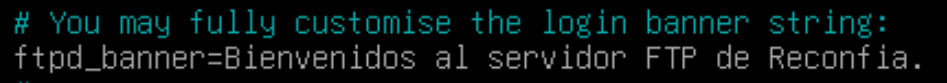
**local\_enable=YES**

**write\_enable=YES**



Establecemos un mensaje de bienvenida al loguearse en el servidor, el mensaje es “Bienvenidos al servidor FTP de Reconfia”

**ftpd=Bienvenidos al servidor FTP de Reconfia**



Limitamos la velocidad de transferencia a los usuarios locales y al usuario anónimo.

**local\_max\_rate=50k**

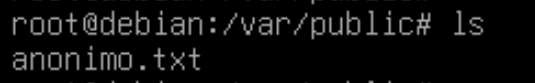
**anon\_max\_rate=50k**

Por defecto el usuario “anonymous” tiene un directorio en la ruta “**/srv/ftp/**”

## 

Lo cambiaremos a la siguiente ubicación “**/var/public**” y crearemos un archivo de prueba, añadimos la siguiente instrucción:

**anon\_root=/var/public *nano anonimo.txt***



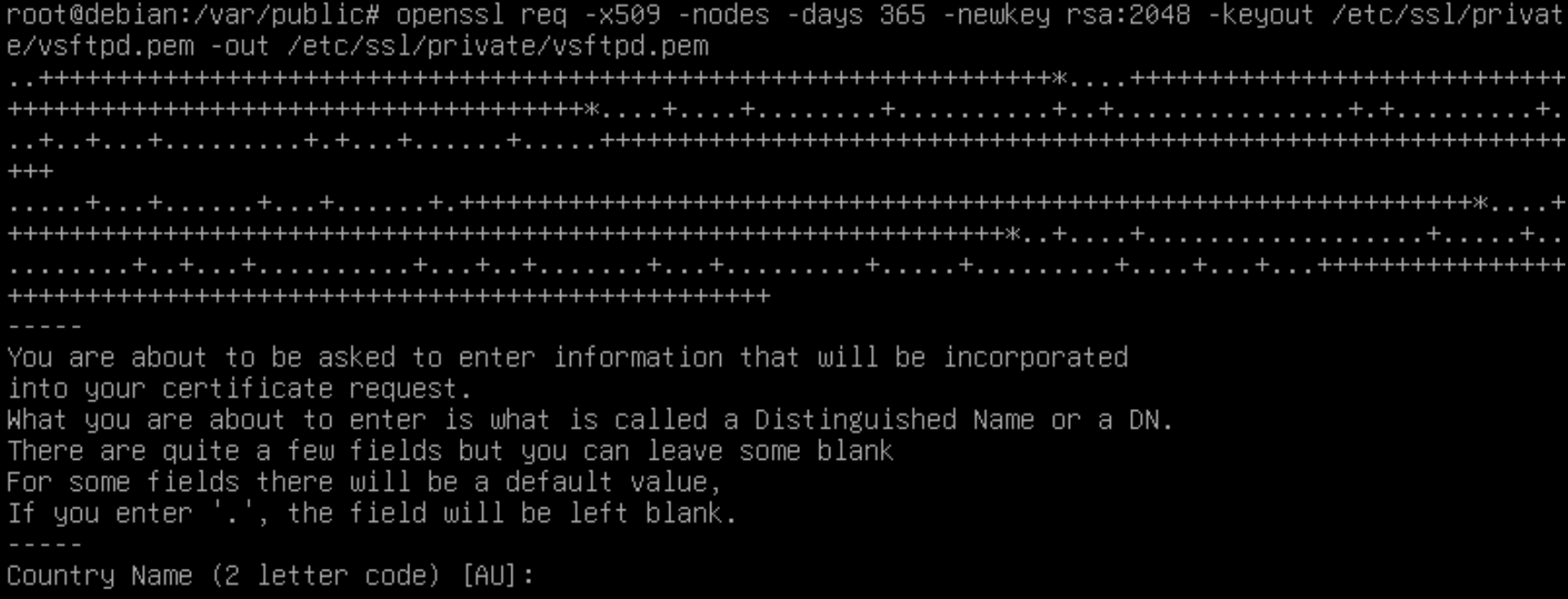
### FTPS

Para habilitar **ftps** en el servidor instalamos OpenSSL para generar un certificado autofirmado X.509, que puede ser utilizado para configurar conexiones seguras SSL/TLS en servicios como el servidor FTP

***sudo apt install openssl***

Entonces generamos el certificado autofirmado.

***sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/vsftpd.pem -out /etc/ssl/private/vsftpd.pem***

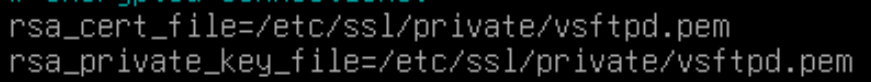


Este comando crea un par de llaves privadas y un certificado autofirmado utilizando el algoritmo RSA con una longitud de clave de 2048 bits. Ambos se almacenan en el mismo archivo (vsftpd.pem) en el directorio /etc/ssl/private/.

Especificamos esa ruta en nuestro archivo vsftpd.conf

**rsa\_cert\_file=/etc/ssl/private/vsftpd.pem**

**rsa\_private\_key\_file=/etc/ssl/private/vsftpd.pem**



Habilitamos ssl

**ssl\_enable=YES**

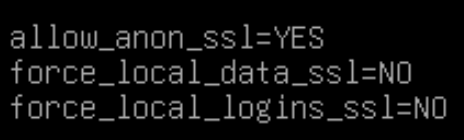


Estas directivas permiten la conexión FTP sin ssl y también permiten ssl con el usuario anónimo.

**allow\_anon\_ssl=YES**

**force\_local\_data\_ssl=NO**

**force\_local\_logins\_ssl=NO**

****

Estos serían los parámetros de configuración esenciales para que el servicio VSFTPD esté en correcto funcionamiento. Por último reiniciamos el servicio para aplicar todos los cambios

***sudo systemctl restart vsftpd.***

### 

### SFTP

SFTP (Secure File Transfer Protocol) es un protocolo seguro utilizado para transferir archivos de forma segura a través de una red. SFTP se ejecuta sobre SSH (Secure Shell) y proporciona cifrado de extremo a extremo, autenticación segura y transferencia segura de archivos. A diferencia de FTP tradicional, SFTP no requiere un servidor separado; en cambio, utiliza el servidor SSH para autenticación y establecimiento de la conexión.

Para instalar un servidor SFTP en sistemas basados en Linux, la solución más común es utilizar OpenSSH, que ya incluye soporte para SFTP.

***sudo apt-get install openssh-server***  
  


Comprobamos el estado del servicio

***s***

### 

## 5º Pruebas

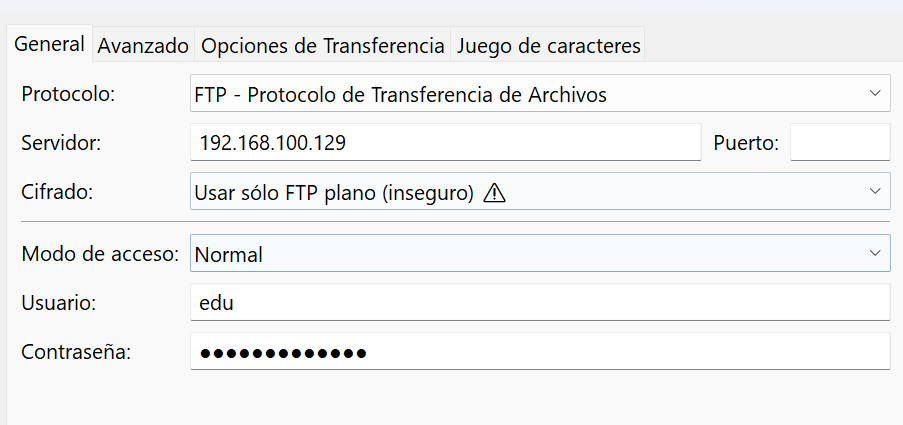
Para hacer la demostración podemos hacerlo mediante clientes ftp como “Filezilla”, “WinSCP”, “CMD” o con el explorador de archivos de Windows.

### Filezilla

Descargamos filezilla en nuestro dispositivo cliente ftp desde la página oficial [Download FileZilla Client for Windows (64bit x86)](https://filezilla-project.org/download.php?platform=win64)

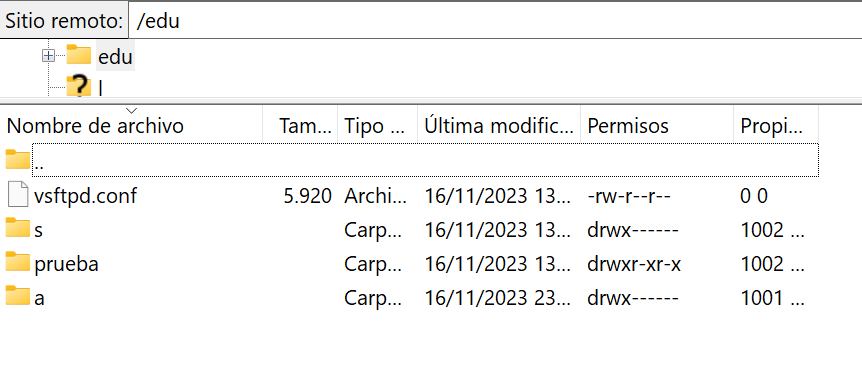
#### FTP plano

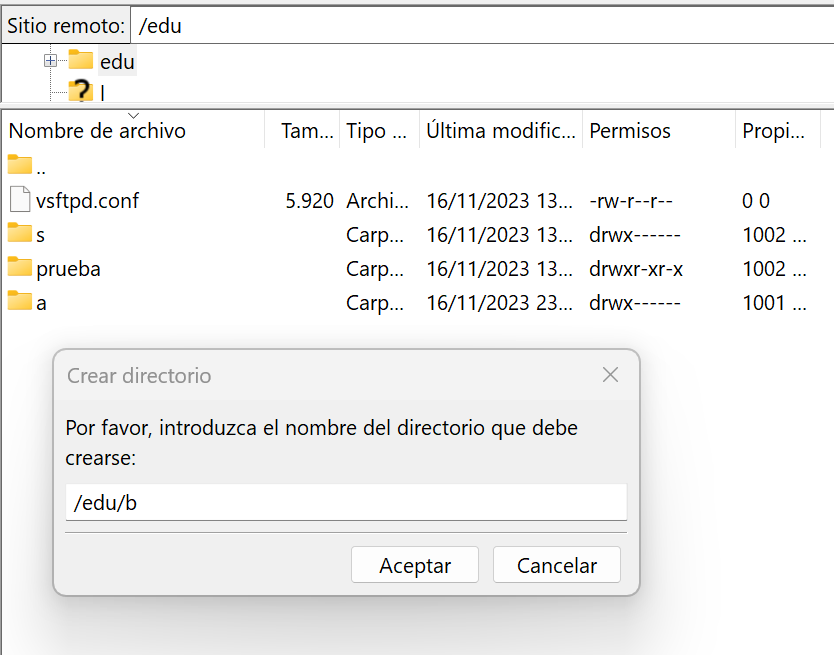
Instalamos y ejecutamos el programa, para crear una conexión **FTP** con nuestro servidor abrimos el Gestor de sitios:

Rellenamos los campos con los datos de nuestro servidor.   
En el protocolo elegimos “ftp”.  
Como servidor introducimos la dirección IP del servidor ftp.

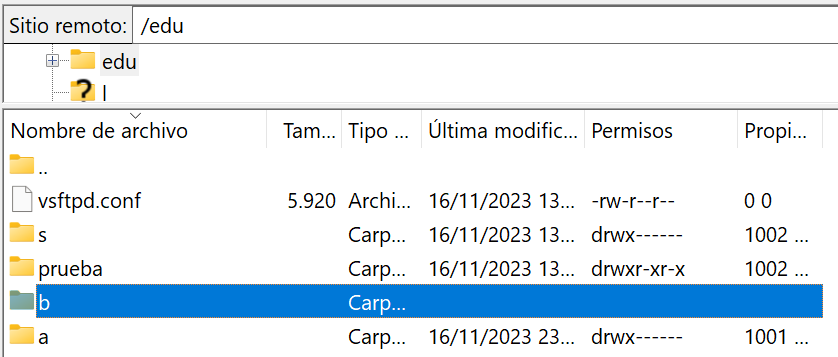
En el cifrado usaremos FTP plano que es inseguro aunque probaremos los otros más adelante.  
En el modo de acceso haremos el acceso normal y luego el anónimo.

Introducimos el usuario y contraseña de nuestra cuenta en el servidor FTP y conectamos.

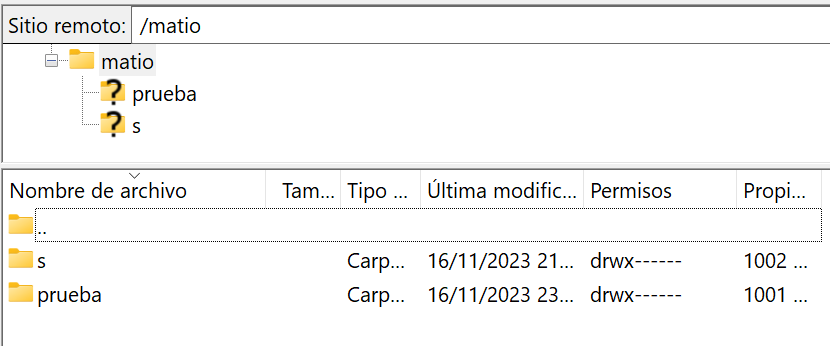




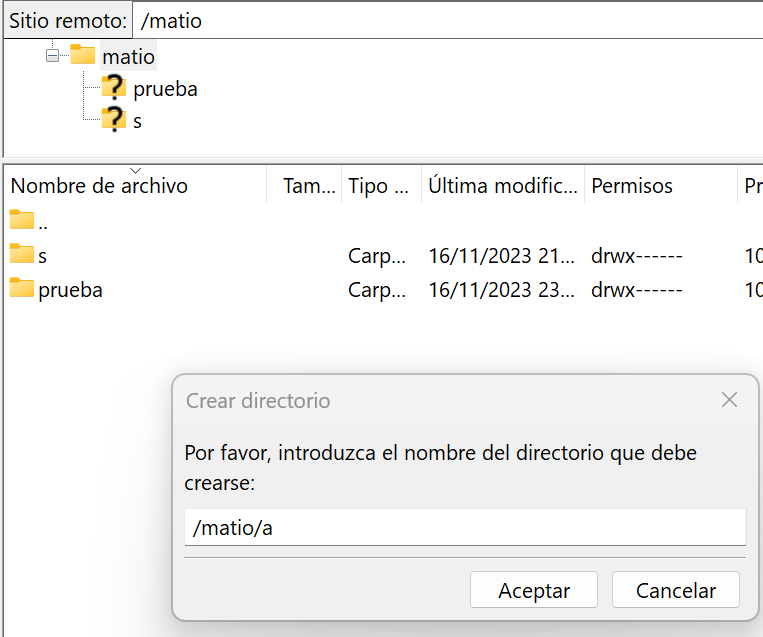
Hacemos una prueba de escritura con el usuario “edu” en su directorio.

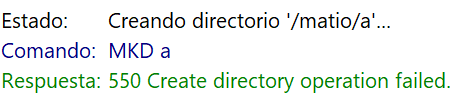


El usuario “edu” es profesor y puede acceder al directorio de los alumnos.

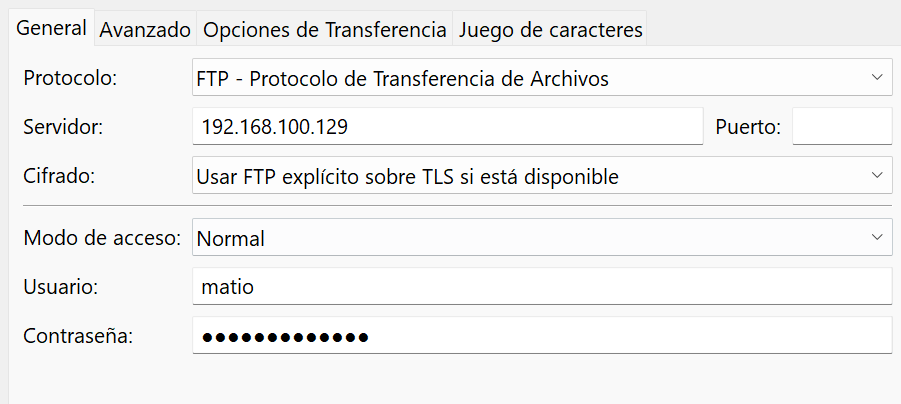


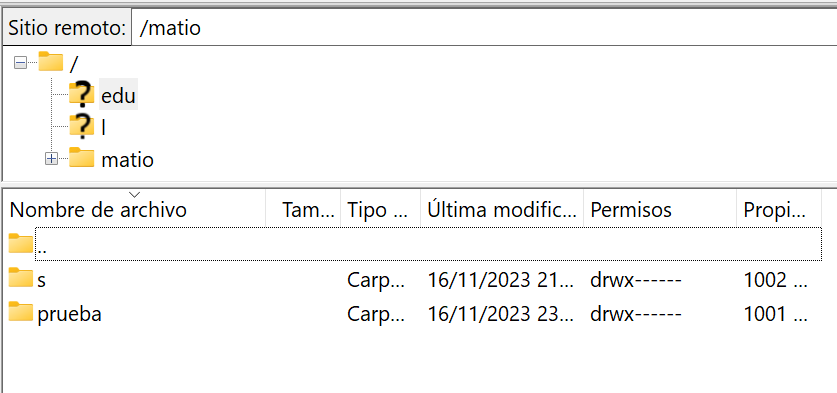
Aunque no tiene permisos de escritura, al crear una carpeta en el directorio del usuario “matio” nos dará un error.



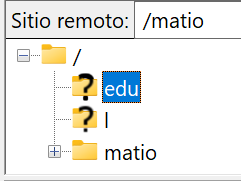


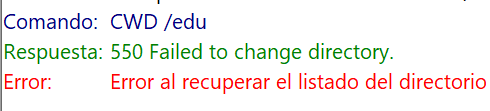
Ahora haremos la prueba con el usuario “matio”.





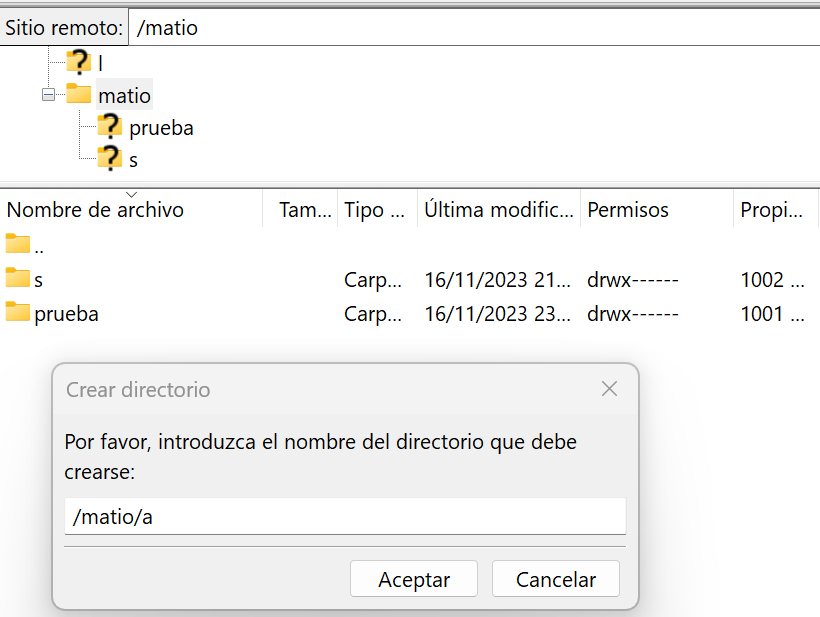
Aquí intentaremos cambiar de directorio y entrar al directorio de “edu”

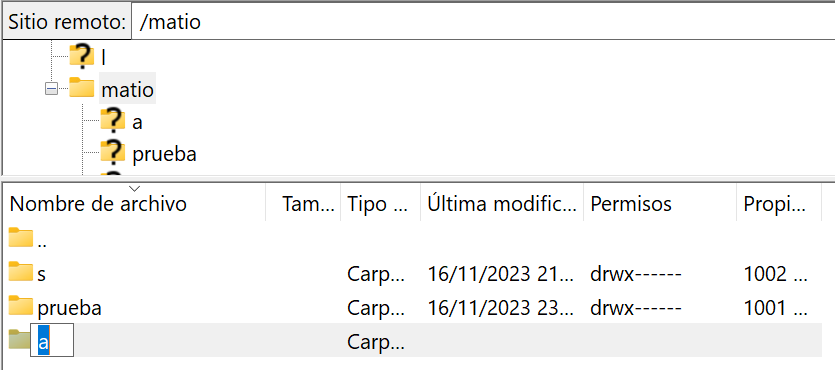




Recibimos el mensaje de error que nos impide cambiar de directorio.

Comprobamos que “matio” puede escribir en su directorio.



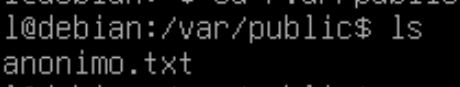


El funcionamiento es el esperado “edu” puede subir y bajar archivos en su propio directorio al igual que “matio” aunque “edu” también puede ver el directorio de “matio” y “matio” solo puede entrar al suyo propio.

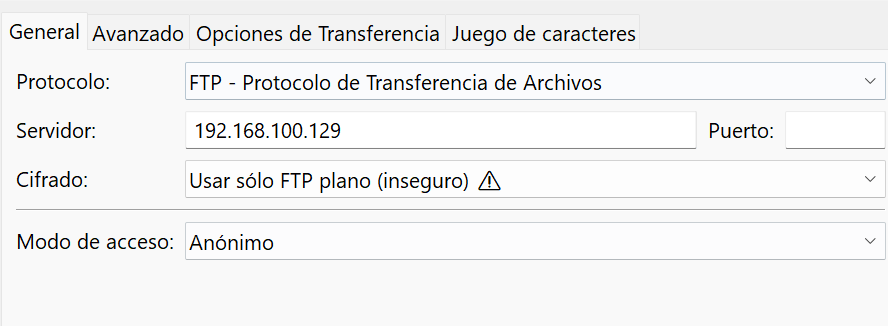
Probaremos el login con el usuario anónimo con un directorio raíz modificado utilizando la siguiente instrucción:

***anon\_root=/var/public***

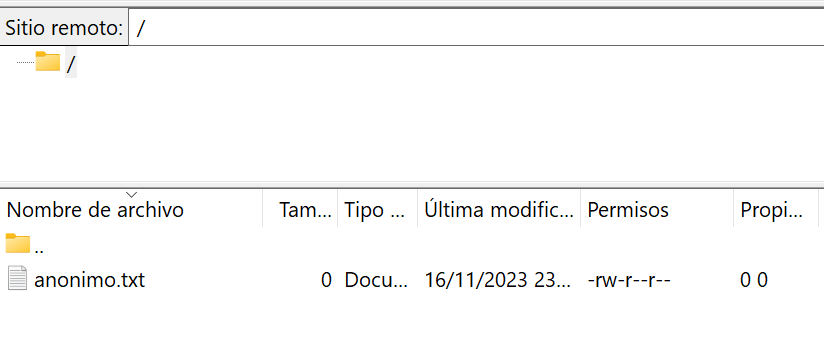
  
  
Dentro de ese directorio creamos un archivo para después encontrarlo al iniciar sesion con el usuario “anónimo”.



Cambiamos el “Modo de acceso” a “anónimo” y conectamos.



Así demostramos que el usuario anónimo está habilitado y que tiene un directorio raíz diferente al de por defecto. Este usuario no puede escribir de forma predeterminada.

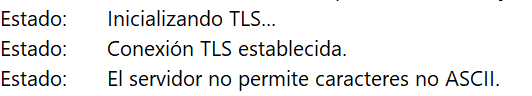
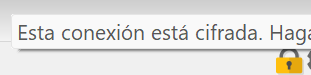


#### FTPS

Para entrar con FTPS lo único que tenemos que cambiar es el cifrado en el gestor de sitio.  
Pasamos de FTP plano a FTP explícito sobre TLS





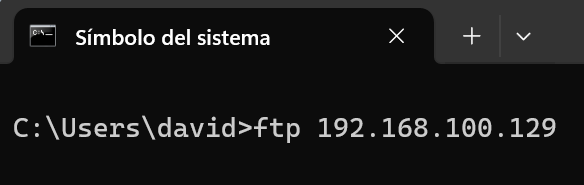
  


Con esto demostramos que la conexión FTPS ha sido establecida.

### CMD

Para iniciar por CMD escribimos el comando ftp y la dirección IP del servidor FTP en la consola de Windows.

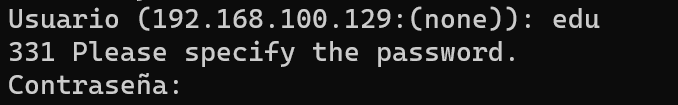
***ftp 192.168.100.129***

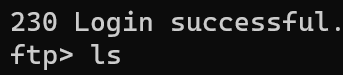




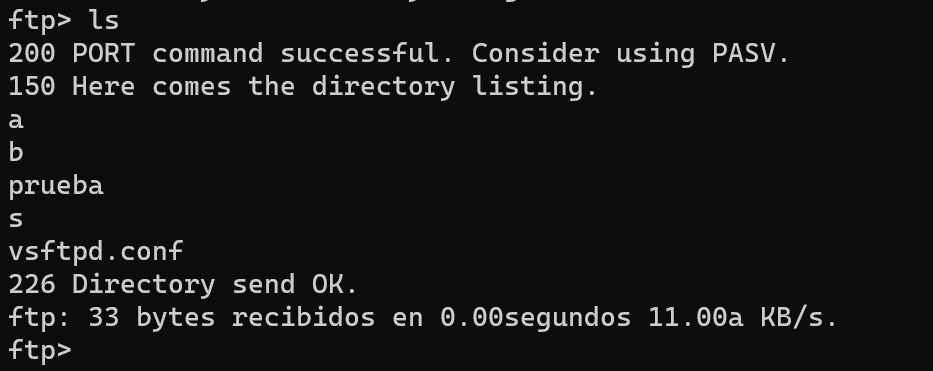
Nos da el mensaje de bienvenida que escribimos en el archivo de configuración.

Completamos el login con nuestro usuario y contraseña





Listamos el directorio de “edu” y observamos su contenido.

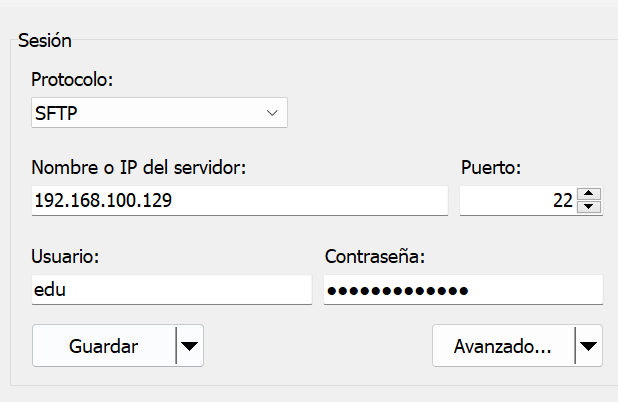


### WinSCP

(Windows Secure Copy) es un cliente de transferencia de archivos gratuito y de código abierto para Windows. Su principal función es permitir la transferencia segura de archivos entre un cliente y un servidor a través de los protocolos SCP (Secure Copy Protocol) y SFTP (SSH File Transfer Protocol).

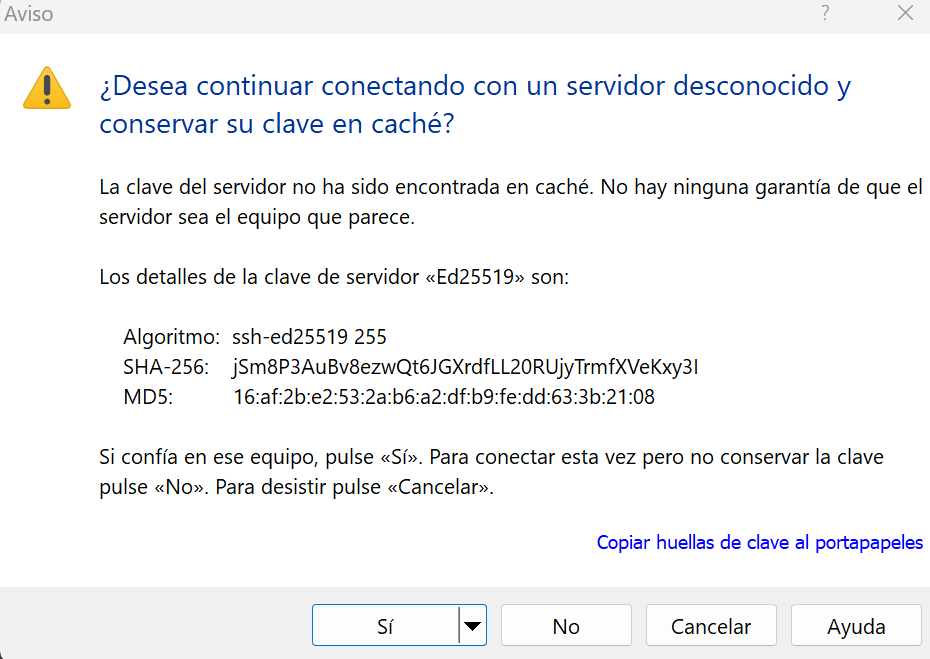
Descargamos WinSCP en nuestro dispositivo cliente FTP desde la página oficial [WinSCP :: Official Site :: Download](https://winscp.net/eng/download.php)

Iniciamos sesión de la misma manera que en Filezilla.

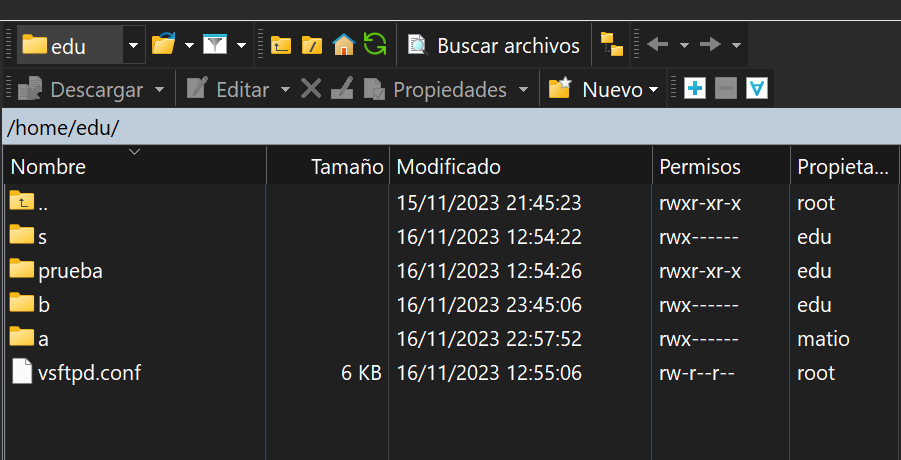


En el protocolo lo cambiamos a SFTP y completamos todos los campos con la IP del servidor, el usuario y la contraseña.

Nos avisa con un mensaje sobre la clave del servidor que no ha sido encontrada en caché  
Este es un mensaje común en el primer inicio de sesión.



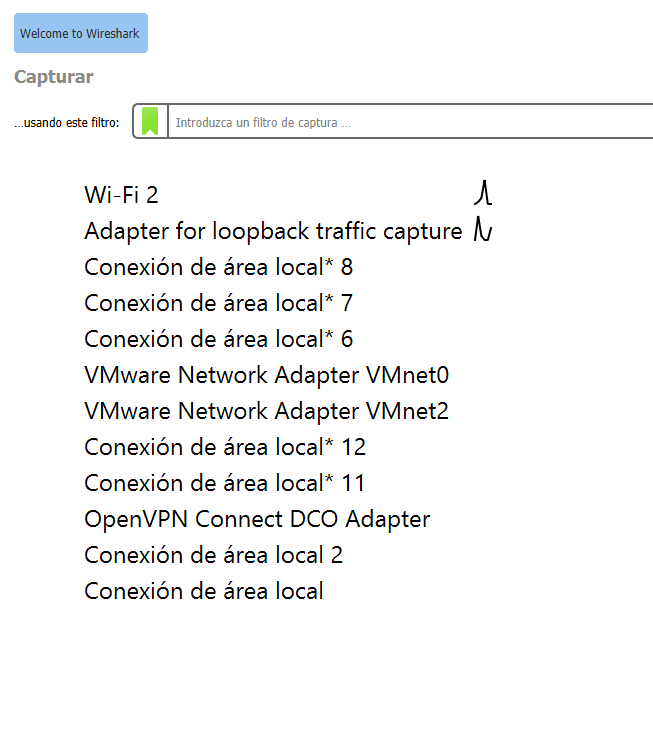
Conectamos con el servidor correctamente



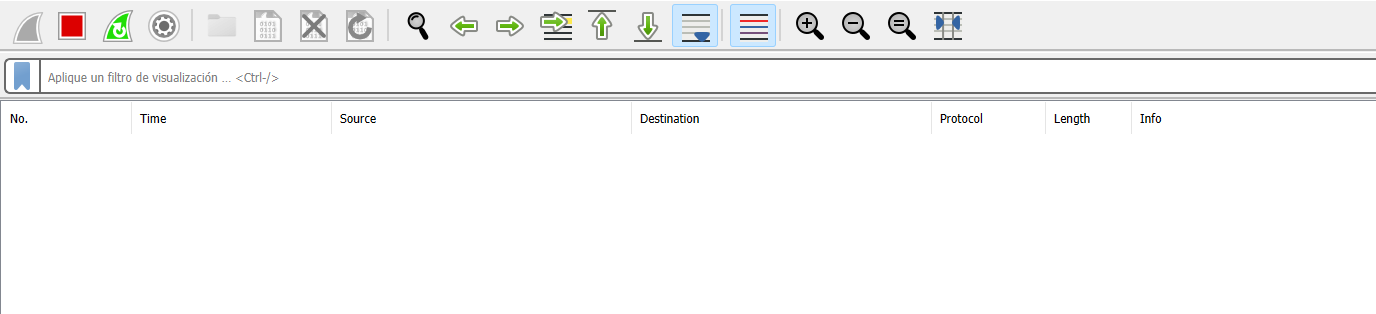
### Wireshark

Wireshark es una herramienta de análisis de red de código abierto que permite capturar y analizar el tráfico de red en tiempo real. Lo usaremos para capturar paquetes entre el servidor FTP y el cliente FTP.

Descargamos Wireshark del sitio oficial [Download Wireshark](https://www.wireshark.org/download.html)



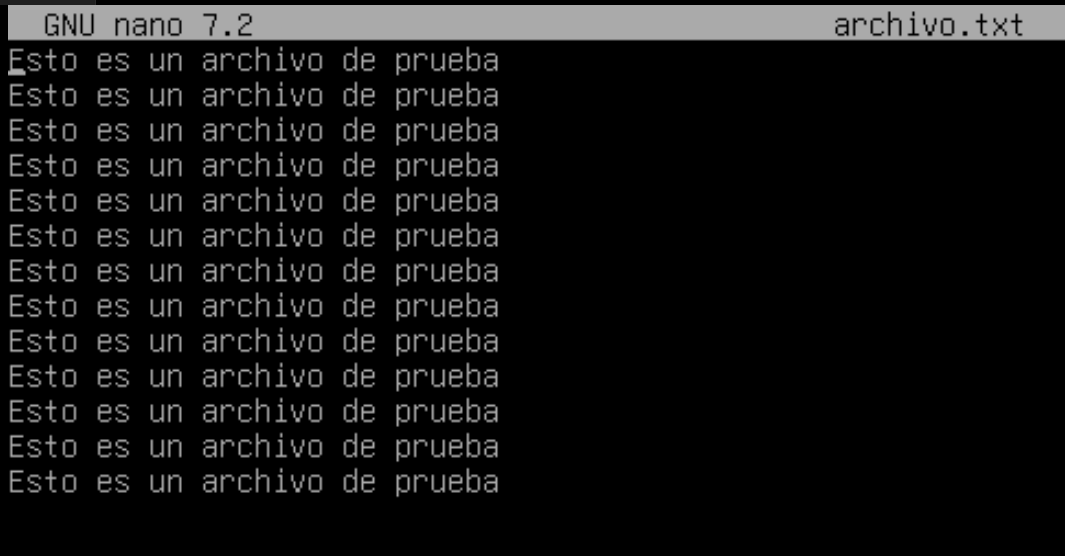
Una vez abierto seleccionamos la interfaz de red que deseamos analizar, en mi caso al utilizar máquinas virtuales de VMware en NAT, elijo la **VMware Network Adapter VMnet2**

****

No tiene una cantidad de tráfico considerable, haremos una transferencia de archivos desde el servidor FTP al cliente para ello utilizaremos Filezilla, después pararemos la captura de paquetes.

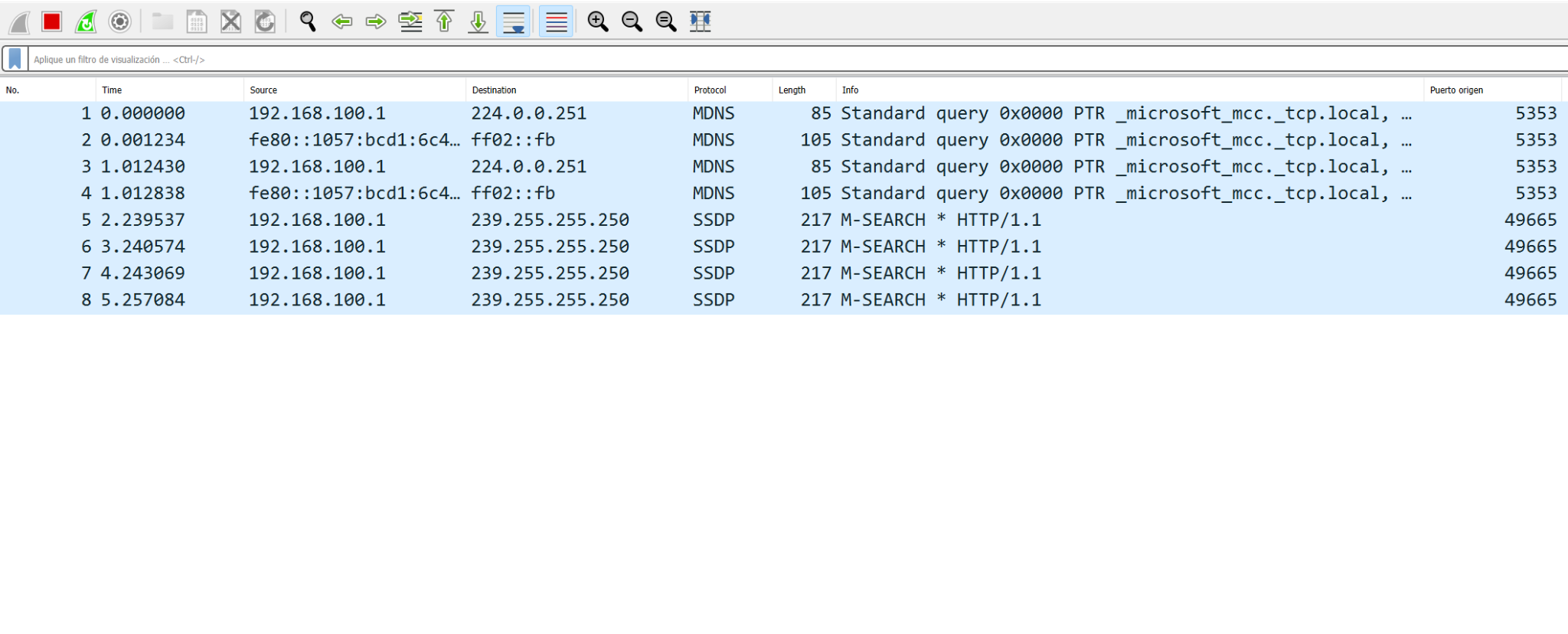
Creamos el archivo que vamos a descargar desde el cliente FTP.

**sudo nano archivo.txt**

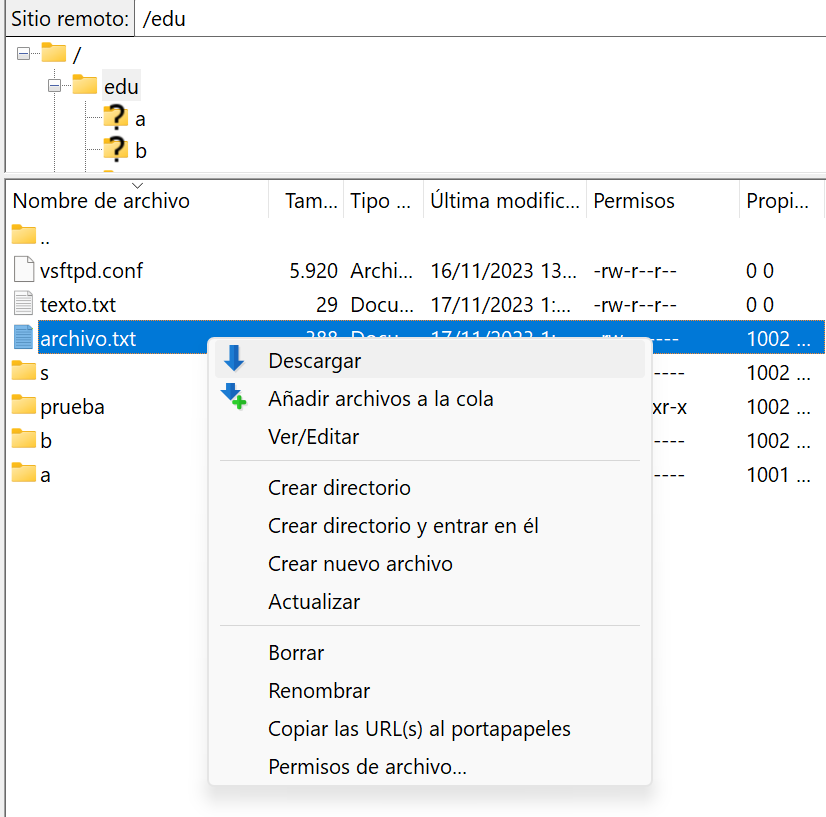


Introducimos algún texto para comprobar que se trata de nuestro archivo cuando lo busquemos con Wireshark.

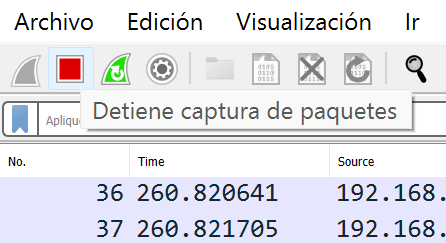
Iniciamos la captura de paquetes. Al principio no veremos paquetes FTP hasta usemos el servicio FTP



Iniciamos sesión en nuestro servidor FTP y descargamos el archivo previamente creado.

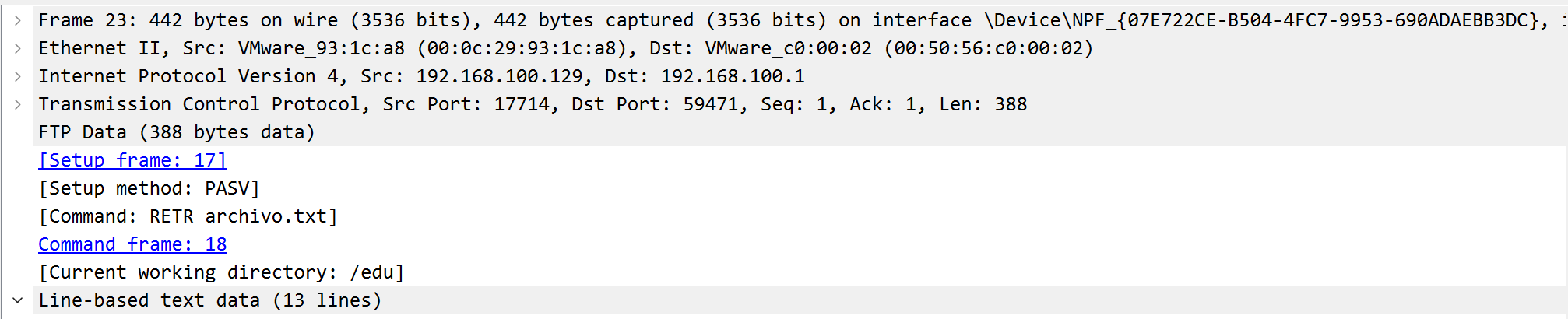


Paramos la captura de paquetes.

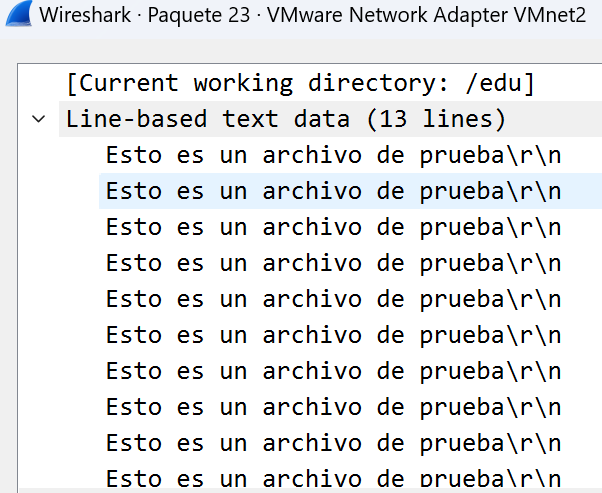
  


Aplicamos el filtro **ftp-data** sobre la captura.

Comprobamos que uno de los paquetes filtrados es el archivo “archivo.txt”.



Su contenido es el que hemos escrito. Esto es posible ya que el protocolo usado es FTP plano sin cifrado, los archivos pueden ser capturados e inspeccionados fácilmente.



## Conclusión

El servidor vsftpd es una solución robusta y segura para ofrecer servicios de FTP en nuestro servidor Debian. Al habilitar los protocolos FTP, FTPS (FTP sobre TLS/SSL) y SFTP, se proporciona flexibilidad y opciones de seguridad para las transferencias de archivos.

### Aspectos Positivos:

#### 1. Seguridad:

- **VSFTPD** es conocido por su enfoque en la seguridad. La configuración predeterminada incluye medidas de seguridad como el modo "chroot" que limita a los usuarios a sus directorios de inicio.

#### 2. Rendimiento:

- **VSFTPD** está diseñado para ser rápido y eficiente, lo que lo hace adecuado para entornos de alto rendimiento.

#### 3. Protocolo FTPS:

- La capacidad de habilitar FTPS proporciona una capa adicional de seguridad mediante el cifrado de las comunicaciones entre el cliente y el servidor.

#### 4. Compatibilidad con SFTP:

- Aunque SFTP no está directamente relacionado con **VSFTPD** (ya que es un protocolo diferente basado en SSH), la inclusión de esta opción permite la elección entre FTP y SFTP según los requisitos específicos de seguridad.

#### 5. Configuración Personalizada:

- La flexibilidad en la configuración permite adaptar el servidor a las necesidades específicas del entorno, incluida la personalización de la bienvenida, la limitación de la velocidad de transferencia, etc.

### Consideraciones y Mejoras Potenciales:

#### 1. Documentación:

- Es importante proporcionar documentación clara y detallada sobre la configuración específica, especialmente para usuarios nuevos o administradores que no estén familiarizados con vsftpd.

#### 2. Monitoreo y Registro:

- Implementar un sistema de monitoreo y registro robusto para detectar posibles problemas o actividades no autorizadas. Revisar regularmente los registros del servidor es esencial para mantener la seguridad.

#### 3. Gestión de Usuarios:

- La gestión eficiente de usuarios, grupos y permisos es esencial. Un sistema centralizado de autenticación y autorización puede facilitar la administración de usuarios en entornos más grandes.

**VSFTPD** es una opción sólida para implementar servicios de **FTP**, y su capacidad para admitir protocolos seguros como **FTPS** y **SFTP** lo hace versátil en términos de requisitos de seguridad. La implementación exitosa dependerá de una configuración cuidadosa y de las prácticas de seguridad implementadas en el entorno específico.