

# SERVICIO DE TRANSFERENCIA DE ARCHIVOS

---

## 1. SERVICIO DE TRANSFERENCIA DE ARCHIVOS

El servicio de transferencia de archivos tiene un papel fundamental a la hora de desplegar una aplicación. Su función es transferir la información de desarrollo a producción.

En el mundo tecnológico existen distintos servidores FTP. Existen también varios modos de conexión, como son el activo y el pasivo, que van a depender de si existe un cortafuegos en mitad de la conexión o no.

Por otro lado existen diferentes tipos de usuarios que se pueden habilitar en este tipo de servicio: autenticados, virtuales y anónimos.

Es un protocolo que permite la transferencia de archivos entre sistemas conectados entre sí. Un cliente tiene la posibilidad de conectar a un servidor para descargar archivos desde él o para enviarle sus propios archivos, independientemente del sistema operativo.

La transferencia de archivos entre cliente y servidor puede ser de todo tipo. El interfaz de transferencia puede ser mediante comandos o modo gráfico.

Este servicio se basa en una arquitectura cliente-servidor, siendo el cliente quien solicita la conexión y el servidor es el que ofrece o almacena los archivos dependiendo de la solicitud del cliente.

El servicio FTP está orientado a conexión que necesita establecer una conexión para poder transferir archivos.

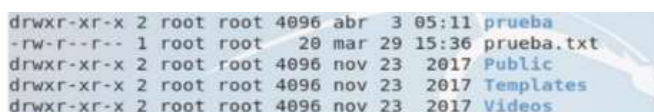
El servidor FTP funciona a través de los siguientes puertos configurables:

- Puerto 21: control de la conexión.
- Puerto 20 o mayor de 2014: puerto de transferencia de datos.

## 2. PERMISOS Y USUARIOS

Los permisos en Linux nos permiten establecer los derechos de acceso a cada archivo o directorio.

Cuando se lista el contenido de un directorio con `ls -l` (o `ll`) aparece la siguiente información:



```
drwxr-xr-x 2 root root 4096 abr  3 05:11 prueba
-rw-r--r-- 1 root root   20 mar 29 15:36 prueba.txt
drwxr-xr-x 2 root root 4096 nov 23  2017 Public
drwxr-xr-x 2 root root 4096 nov 23  2017 Templates
drwxr-xr-x 2 root root 4096 nov 23  2017 Videos
```

El primer carácter identifica el tipo de fichero:

d: directorio

- : fichero

l: enlace

b: archivo binario

c: archivo de caracteres especiales

Los permisos están compuestos por tres letras (r, w, x), y hay tres grupos de permisos, uno por cada tipo de usuario.

- Primer rwx: permisos del usuario propietario
- Segundo rwx: permisos del grupo (usuarios que pertenecen a ese grupo)
- Tercer rwx: permisos de los demás (otros, todos los demás usuarios)

Si hay un guión en la posición de un permiso quiere decir que ese permiso no se tiene. Los permisos quieren decir cosas distintas dependiendo de si se aplica a un archivo o a un directorio.

#### **Si se aplican a un archivo:**

r: Lectura. Se puede ver el contenido del archivo (y se puede copiar).

w: Escritura. Se puede modificar el contenido del archivo. ojo: este permiso no es el que dice si se puede borrar el archivo o renombrar. Eso depende del directorio que lo contiene.

x: Ejecución. Se puede ejecutar el archivo. eso quiere decir que el archivo es ejecutable (es un programa o un script).

#### **Si se aplican a un directorio:**

rx: Se puede entrar en el directorio pero no se puede modificar el contenido del directorio. Los permisos "r" o "x" no se suelen poner aislados, ya que por sí solos no permiten la entrada en el directorio.

rwx: Se puede modificar el listado del directorio. quiere decir que se pueden borrar o renombrar los archivos del directorio, y además permite crear y pegar archivos dentro del directorio.

wx: Se pueden pegar archivos en el directorio, pero no se puede entrar en él. Esto es útil para crear una especie de directorio buzón en el que los demás usuarios puedan pegar archivos en un directorio sin ver lo que ya hay en ese directorio.

## **CAMBIAR PERMISOS**

### **chmod ugoa +/- rwxst**

u: usuario

g: grupo

o: los demás

a: cambiar a usuario, grupo y los demás

+ : añadir

- : eliminar

=: asignar permisos absolutos

## NOTACIÓN OCTAL

Para poner permisos absolutos se suele usar la notación octal:

Número decimal	Binario	Permisos efectivos
0	000	---
1	001	--x
2	010	-w-
3	011	-wx
4	100	r--
5	101	r-x
6	110	rw-
7	111	rwX

### Ejemplo:

Un archivo con los siguientes permisos `rw-r---wx`, en notación octal tendría los siguientes permisos: `643`

## CAMBIAR PROPIETARIO

### **chown [opciones] [usuario][:grupo] ficheros**

Por ejemplo para cambiar el usuario y el grupo del archivo prueba a daw2 que pertenece al grupo daw2 sería de la siguiente manera:

```
chown daw2:daw2 prueba.txt
```

## USUARIOS Y GRUPOS

- **adduser**  
adduser [opciones] nombre\_usuario  
*Opciones:*  
--ingroup nombre\_grupo → Añade el usuario
- **addgroup**  
addgroup nombre\_grupo  
Al crear el grupo se añade al fichero group
- **usermod**  
Modificar las opciones de usuario  
usermod [opciones] Nombre\_usuario
- **deluser**  
Borrar usuarios  
deluser nombre\_usuario
- **delgroup**  
Borrar grupos  
delgroup nombre\_grupo

### 3. TIPOS DE USUARIOS

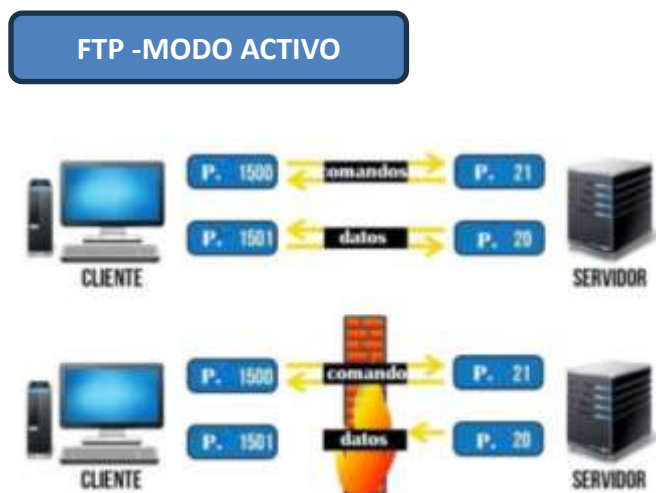
**Usuarios locales:** usuarios con cuenta de usuario en el sistema servidor. Inician sesiones FTP identificándose con el mismo nombre de cuenta y contraseña que usarían en la máquina servidora para iniciar una sesión de sistema operativo. Cuando inician una sesión FTP, su directorio de trabajo remoto es su directorio personal en el sistema (/home/usuario).

**Usuarios anónimos:** son nombres de usuarios que pueden iniciar sesiones FTP sin contraseña pero si identificándose con un nombre. Por defecto vsftpd admite dos usuarios anónimos llamados anonymous y ftp. El usuario anónimo ftp puede cambiarse en la configuración. Los usuarios anónimos tienen como directorio remoto FTP el directorio /srv/ftp. Este directorio puede cambiarse con la directiva anon\_root.

**Usuarios virtuales:** son usuarios que no tienen cuentas de usuario en el sistema servidor pero que pueden acceder al servicio FTP. Acceden todos a través de una misma cuenta de usuario del sistema. Cuando haya usuarios virtuales, se puede establecer una configuración de acceso para cada uno de ellos.

Las directivas del archivo de configuración /etc/vsftpd.conf se pueden modificar para configurar el acceso al servicio por los usuarios.

### 4. MODOS DE CONEXIÓN DEL CLIENTE: ACTIVO Y PASIVO



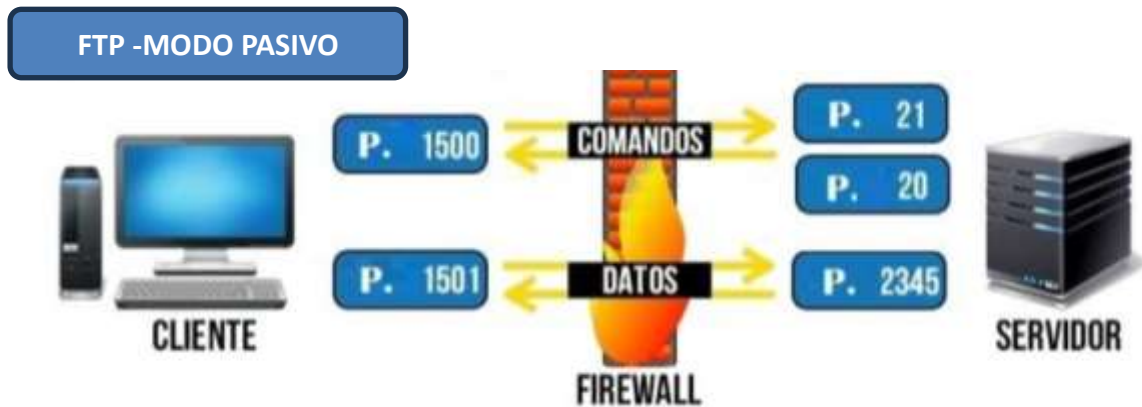
Este modo funciona cuando el cliente solicita el servidor, a través de un puerto aleatorio, con un paquete dirigido al puerto 21. Una vez establecida la conexión, el servidor inicia otra.

El servidor, a través del puerto 20, se pone en contacto inmediatamente con el puerto siguiente del cliente. Si el puerto utilizado en la primera conexión fue el 1500, el utilizado en la segunda conexión será el 1501.

Una vez establecida la conexión, todas las transferencias de archivos se realizan a través de los mismos puertos entre el cliente y el servidor. Por lo tanto, el cliente establece el canal de comandos, pero es el servidor que establece el canal de datos.

En el segundo esquema, la presencia de un firewall bloquea el intento de comunicación entre servidor y cliente, ya que el servidor utiliza un puerto diferente de la primera conexión. Los equipos con cortafuegos rechazan estas conexiones aleatorias porque es el servidor quien las inicia por un puerto que no estará abierto.

Este es el principal problema del FTP modo activo.



Es el cliente quien inicia la conexión con el servidor para evitar bloqueos de conexión mediante cortafuegos. El cliente inicia ambas conexiones, control y datos. Si no existiera cortafuegos no habría problema. Pero al existir cortafuegos, el servidor, devuelve la respuesta por un puerto diferente, que hace que el cortafuegos bloquee la conexión.

## 5. INSTALACIÓN FTP SERVIDOR Y CLIENTE (Prácticas de FTP)

### SERVIDOR FTP EN LINUX

#### ACCESO CON USUARIO ANÓNIMO

- 1) `sudo apt-get install vsftpd`
- 2) Configuración del servidor.

En /etc editar el fichero `vsftpd.conf`

En el archivo se realizan las modificaciones necesarias para restringir permisos a los usuarios\*

- 3) Reiniciar el servicio.

`sudo service vsftpd restart`

### CLIENTE FTP EN LINUX

Instalar Cliente FTP: `sudo apt-get install filezilla`

#### Acceso con comandos:

- acceder: `ftp ip_servidor`
- acceder con un determinado usuario: `ftp ip_servidor:nombre_usuario`
- descargar fichero: `get nombre_fichero`
- enviar un archivo al servidor: `put nombre_fichero`
- borrar una archivo del servidor: `delete nombre_fichero`
- salir: `quit`

## \* RESTRINGIR PERMISOS A LOS USUARIOS

Modificar los siguientes parámetros en el archivo vsftpd.conf

- **listen = YES:** Para que se inicie con el sistema.
- **anonymous\_enable = NO:** No permitimos que usuarios anónimos puedan conectarse a nuestro servidor. Es por seguridad.
- **local\_enable = YES :** Para poder conectarse con los usuarios locales del servidor donde está instalado.
- **write\_enable = YES :** Si quieres que los usuarios puedan escribir y no sólo descargar cosas.
- **local\_umask = 022 :** Esta máscara hace que cada vez que subas un archivo, sus permisos sean 755. Es lo más típico en servidores FTP. (Si no está en el archivo de configuración se puede cambiar)
- **chroot\_local\_user = YES:** Si está activada, a los usuarios locales se les cambia el directorio raíz
- **chroot\_list\_enable = YES:** Sirven para que los usuarios locales puedan navegar por todo el árbol de directorios del servidor. Evidentemente esto sólo queremos permitirlo a ciertos usuarios, para ello tenemos el siguiente parámetro.
- **chroot\_list\_file = /etc/vsftpd.chroot\_list:** Cuando este archivo es usado con la opción “chroot\_list” habilitada, establece una lista de usuarios que serán enjaulados a su directorio home. Estos usuarios no podrán cambiar de directorio fuera de su directorio home.

Después de cada cambio que se haga en el archivo de configuración hay que reiniciar el servicio:

**sudo service vsftpd restart**

## 6. ALTERNATIVAS A FTP PARA COMPARTIR ARCHIVOS

### SMB (SAMBA)

Uno de los protocolos más utilizados en redes domésticas y profesionales es SMB (samba), un protocolo seguro que en su versión SMB 3.0 incorpora autenticación y transferencia de datos con cifrado AES, de esta forma, todas las comunicaciones estarán cifradas y autenticadas, proporcionándonos mejor seguridad y privacidad. Las versiones anteriores como SMB 1.0 o SMB 2.0 no lo incorporaban.

Este protocolo está orientado a compartir archivos y carpetas a través de la red local. No debería usarse fuera de esta.

### SFTP

Otro protocolo que podemos usar para transferir archivos y carpetas a través de la red local o Internet es SFTP. Este protocolo está basado en SSH, por tanto, tendremos cifrado de datos tanto en la autenticación como en la transferencia de datos, por lo que es uno de los protocolos más recomendable si lo que quieres es tener la mejor seguridad posible.

### SSH

SSH (Secure SHell, en español: intérprete de órdenes segura) es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red y copiar los datos que residen en ella de forma segura.

Además de la conexión a otros dispositivos, SSH nos permite copiar datos de forma segura, gestionar claves RSA para no escribir claves al conectar a los dispositivos y pasar los datos de cualquier otra aplicación por un canal seguro tunelizado mediante SSH.

#### Seguridad

SSH usa técnicas de cifrado que hacen que la información que viaja por el medio de comunicación vaya de manera no legible y ninguna tercera persona pueda descubrir el usuario y contraseña de la conexión ni los datos que se intercambian durante toda la sesión.

### SSH EN LINUX

#### Instalar en cliente y servidor

```
sudo apt-get install openssh-server
```

Está instalado en /etc/ssh

#### Iniciar servicio

```
sudo service ssh start
```

Los servicios están en /etc/init.d

## Conexión por ssh desde la máquina cliente:

ssh administrador@10.112.0.103

## EN EL SERVIDOR:

Podemos modificar el archivo de configuración de sshd el cual se encuentra en la ruta **/etc/ssh/sshd\_config**

- **AllowUsers (añadirlo al final del fichero)**

Aquí podemos permitir el acceso vía SSH a usuarios particulares. Si vamos a especificar múltiples usuarios lo separamos por espacios.

AllowUsers usuario@10.112.0.\* (Rango)

AllowUsers usuario1@10.112.0.101 (ip de la máquina a la que se permite acceder)

- **Cambiar banner de bienvenida:** Banner /etc/issue.net

## MEDIDAS PARA MEJORAR LA SEGURIDAD POR SSH

### 1. Cambiar el puerto por defecto

Por defecto, SSH utiliza el puerto 22, por lo que cuando un hacker lanza un ataque lo suele hacer sobre este puerto. Si le cambiamos el número del puerto, el servicio no responderá al puerto por defecto.

*port 9122*

*Para conectar desde el cliente añadir el puerto de acceso: ssh usuario1@10.16.0.103 -p 9122*

### 2. Deshabilitar el acceso root

Para mejorar la seguridad podemos impedir el acceso al servidor por medio de este usuario root y obligar al acceso por medio de alguno de los usuarios que hayamos creado y que no tienen privilegio root.

*PermitRootLogin no*

### 3. Limitar el número de reintentos

Podemos indicar el número de veces que nos podemos equivocar al introducir el nombre de usuario o la contraseña.

*MaxAuthTries 3*

### 4. Limitar el tiempo que estará disponible la pantalla de login

Indicamos el tiempo en segundos que estará disponible la pantalla de login para introducir nuestras credenciales. Pasado ese tiempo, la pantalla desaparecerá y habrá que volver a iniciar el proceso.

*LoginGraceTime 20*

Después de hacer cambios en el archivo de configuración, reiniciar el servicio.