

Servidores FTP

Normalmente para subir archivos en Internet, ya sean de texto, imágenes, vídeo... hubo que emplear algún método de transferencia de archivos para ubicarlos.

Uno de los métodos más empleados como servicio de transferencia de archivos se realiza mediante el servicio ftp. Éste utiliza el protocolo FTP empleando la arquitectura cliente-servidor. Así el servidor ftp esperará peticiones para transferir los archivos y el cliente ftp, ya sea por terminal o de modo gráfico, realizará esas peticiones.

Uno de los principales problemas, a pesar de ser uno de los métodos más utilizados del protocolo FTP es la no seguridad de la información, esto es, la transferencia tiene lugar sin cifrar la información transferida. Este no sólo es un problema del protocolo FTP sino de muchos de los protocolos utilizados en Internet, puesto que en el comienzo de Internet no se preveía su expansión actual y no se pensaba en asegurar la información mediante cifrado, sino simplemente asegurar el buen funcionamiento. Hoy en día existen extensiones sobre el protocolo FTP que aseguran el cifrado en la transferencia, como FTPS, empleando el cifrado SSL/TLS.

Funcionamiento

El protocolo FTP emplea una arquitectura cliente/servidor, siendo el cliente ftp quien solicita la transferencia de archivos y el servidor ftp quien ofrece los archivos. Pertenece a la familia de protocolos de red TCP y por lo tanto es un protocolo orientado a conexión, esto es, el cliente ftp necesita establecer una conexión con el servidor para empezar la transferencia de ficheros. Si no se establece la conexión ésta no tiene lugar.

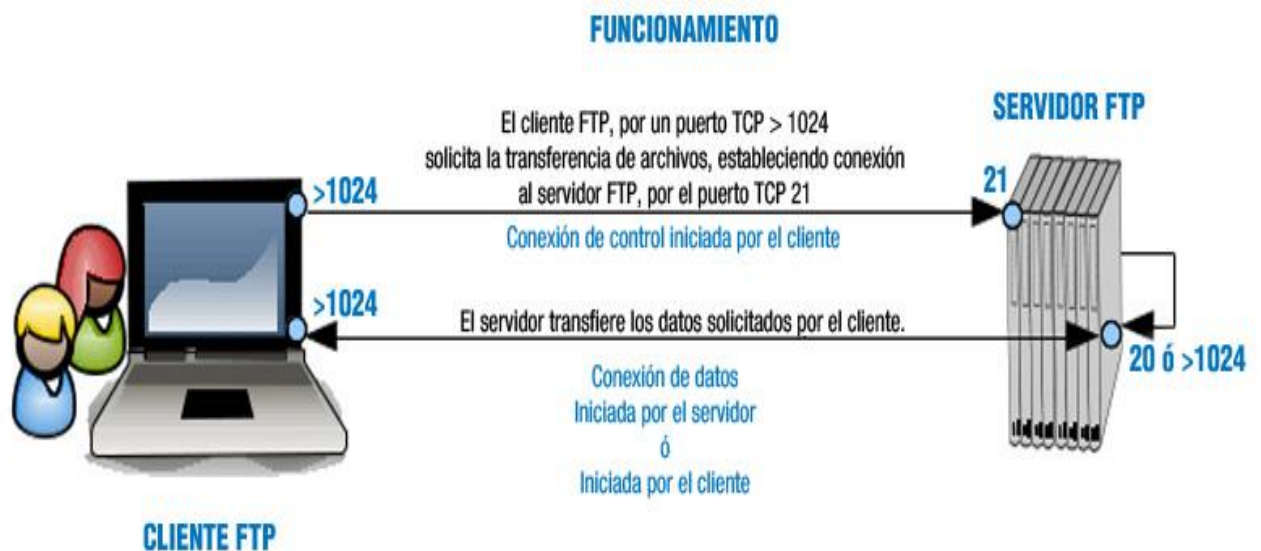
Puesto que FTP es un protocolo que no utiliza una autenticación de usuarios y contraseña cifrada, se considera un protocolo inseguro y no se debería utilizar a menos que sea absolutamente necesario. Verás que existen otras alternativas al FTP, como por ejemplo el protocolo FTPS, para mantener comunicaciones cifradas. Aun así, el protocolo FTP está muy extendido en Internet ya que a menudo los usuarios necesitan transferir archivos entre máquinas sin importar la seguridad.

El protocolo FTP requiere de dos puertos TCP en el servidor para su funcionamiento, a diferencia de la mayoría de los protocolos utilizados en Internet que solamente requieren un puerto en el servidor. Un puerto es necesario para establecer el control de la conexión y otro se utiliza para el control de la transmisión, es decir, un puerto se utiliza para establecer la conexión entre el cliente y el servidor y otro para la transferencia de datos.

Los puertos TCP del servidor en cuestión, suelen ser el 21 para el control de la conexión y otro a determinar según el modo de conexión: podría ser el 20 o incluso uno mayor de 1024. Hay que tener en cuenta que estos puertos pueden ser modificados en la configuración del servidor, así no es

obligatorio que los puertos 21 y 20 sean los asignados al servidor FTP, pero sí son los que éste maneja por defecto. El puerto 21 también es conocido como puerto de comandos y el puerto 20 como puerto de datos.

La ventaja que supone utilizar el protocolo FTP se basa en su alto rendimiento y sencillez, que lo hacen una opción conveniente para la transferencia de archivos a través de Internet.



Para poder establecer una conexión con el protocolo FTP son necesarias dos partes: un servidor y un cliente.

Existen múltiples tipos de clientes ftp, desde clientes en terminal de comandos, como ftp o lftp, clientes gráficos como gftp o FileZilla, hasta un cliente ftp en los navegadores mediante ftp://

Tipos de usuarios

Típicamente existen dos tipos de usuarios:

- Usuarios anónimos: usuarios que tienen acceso y permisos limitados por el sistema de archivos. Al conectarse al servidor FTP sólo deben introducir una contraseña simbólica, normalmente cualquier dirección de correo -real o ficticia-, por ejemplo: a@ .
- Usuarios del sistema: aquellos que disponen de una cuenta en la máquina que ofrece el servicio FTP. Al conectarse al servidor FTP deben introducir su contraseña de sistema.

Pero en ciertos servidores, como el servidor ProFTPD, existe una tercera posibilidad muy interesante: usuarios virtuales. Los usuarios virtuales poseen acceso y permisos al servidor FTP sin necesidad de ser usuarios del sistema, por lo tanto si un usuario virtual quisiera acceder al sistema operativo como si fuese un usuario del sistema, ya sea de forma local o remota no podría, pues su cuenta de usuario no existe en el sistema. Los usuarios virtuales tienen definida una contraseña propia y pueden estar definidos en ficheros de

autenticación (de texto) con el mismo formato que los del sistema operativo GNU/Linux `/etc/passwd`, directorios LDAP, bases de datos SQL y servidores RADIUS.



Dependiendo del servidor ftp, podrás tener unos métodos de autenticación de usuarios u otros, por ejemplo en el servidor ftp ProFTPD se permite los siguientes métodos:

- Ficheros de autenticación del sistema operativo: **`/etc/passwd`** y **`/etc/shadow`**: Para ello usa las directivas **`AuthUserFile`** y **`AuthGroupFile`**.
- Usuarios virtuales definidos mediante ficheros de autenticación (de texto) propios, distintos de los del sistema operativo: para ello también usa las directivas **`AuthUserFile`** y **`AuthGroupFile`**.
- Autenticación PAM Es necesario establecer la directiva **`AuthPAMAuthorative`** a 'on'.
- Bases de datos SQL, tales como MySQL o Postgres. Para ello emplea el módulo `mod_sql`; más información sobre el uso de `mod_sql` lo puedes encontrar en el [HowTo SQL](#)
- LDAP: Para ello emplea el módulo `mod_ldap`.
- RADIUS: Para ello emplea el módulo `mod_radius`.

Mediante la [directiva `UserPassword`](#) se puede crear una contraseña para un usuario particular que sobrescribe la contraseña del usuario en **`/etc/passwd`** (o **`/etc/shadow`**), esta contraseña es solamente efectiva dentro del contexto en el cual la directiva es aplicada, esto es, no se modifica el fichero **`/etc/passwd`** (o **`/etc/shadow`**) sino que se da la posibilidad de que el usuario emplee otra contraseña distinta de la definida en los ficheros del sistema operativo.

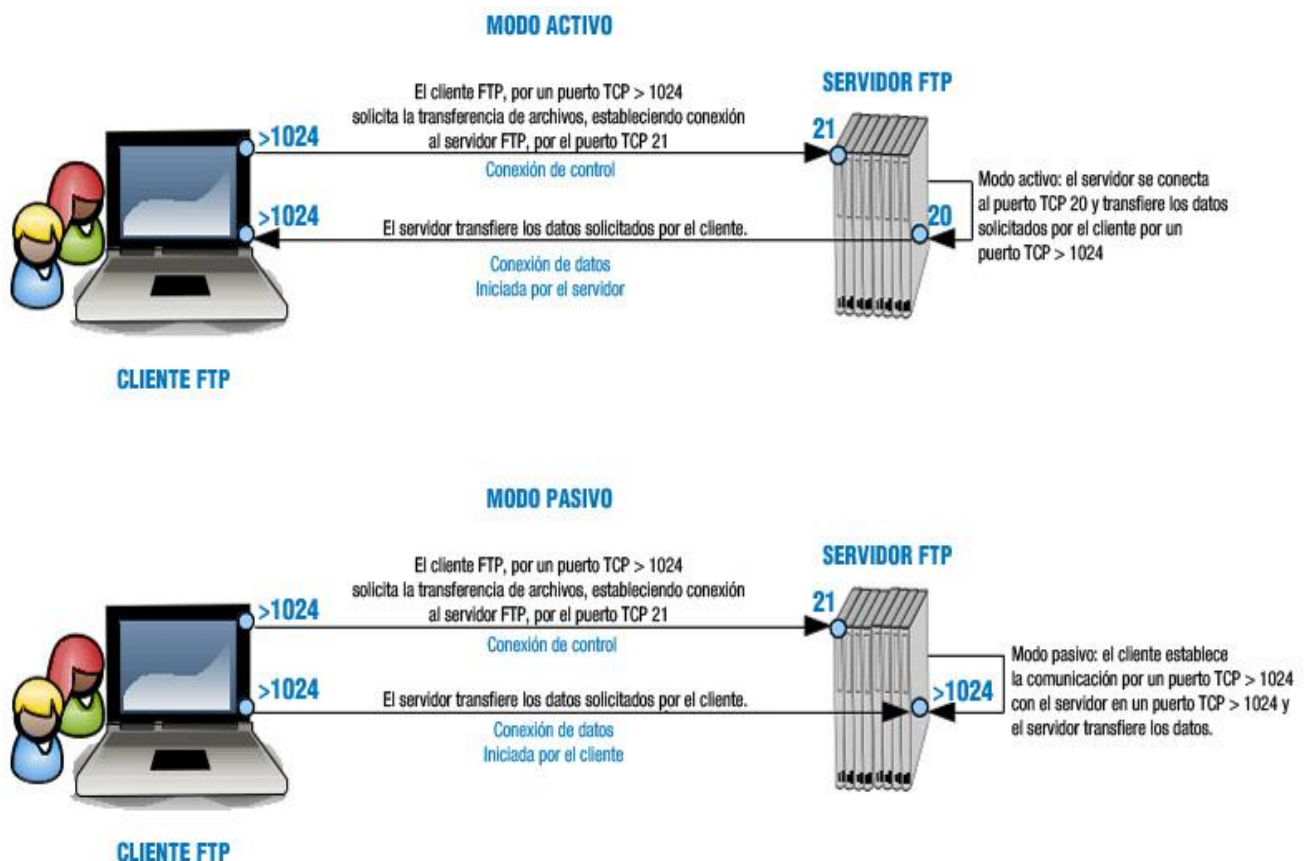
Modos de conexión

El modo de conexión del cliente ftp, que puede ser activo o pasivo.

Cuando una aplicación cliente FTP inicia una conexión a un servidor FTP, abre el puerto 21 en el servidor. Se utiliza este puerto para

arrojar todos los comandos al servidor. Cualquier petición de datos desde el servidor se devuelve al cliente a través de otro puerto TCP del servidor dependiendo del modo de conexión del cliente. Así:

- El modo activo es el método original utilizado por el protocolo FTP para la transferencia de datos a la aplicación cliente. Cuando el cliente FTP inicia una transferencia de datos, el servidor abre una conexión desde el puerto 20 en el servidor para la dirección IP y un puerto aleatorio sin privilegios (mayor que 1024) especificado por el cliente. Este arreglo implica que la máquina cliente debe poder aceptar conexiones en cualquier puerto superior al 1024. Con el crecimiento de las redes inseguras, tales como Internet, es muy común el uso de cortafuegos para proteger las máquinas cliente. Debido a que estos cortafuegos en el lado del cliente normalmente rechazan las conexiones entrantes desde servidores FTP en modo activo, se creó el modo pasivo.
- La aplicación FTP cliente es la que inicia el modo pasivo, de la misma forma que el modo activo. El cliente FTP indica que desea acceder a los datos en modo pasivo y el servidor proporciona la dirección IP y el puerto aleatorio, sin privilegios (mayor que 1024) en el servidor. Luego, el cliente se conecta al puerto en el servidor y descarga la información requerida.



En sistemas GNU/Linux es típico encontrar el archivo `/etc/services` que contiene una lista de puertos TCP/UDP relacionado con los servicios estándar

que trabajan en los mismos. Ejecuta el comando **cat /etc/services | grep ftp** y encontrarás todos los puertos y servidores relacionados con la cadena **ftp**.

Tipos de archivos

Desde el punto de vista de FTP, los archivos se agrupan en dos tipos:

- Archivos ASCII: son archivos de texto plano (.txt, .ps, .html...)
- Archivos binarios: todo lo que no son archivos de texto: ejecutables (.exe), imágenes (.jpg, .png ...), archivos de audio (.mp3, .wav ...), vídeo (.avi, .mov ...) , etcétera.

Es muy importante saber con qué tipo de archivos estás trabajando en la transferencia ya que si no utilizas las opciones adecuadas puedes destruir la información del archivo. El servidor ftp permite configurar la transferencia de archivos según el tipo del mismo, es por eso que al ejecutar el cliente FTP, antes de transferir un archivo, debes utilizar uno de los siguientes comandos o poner la correspondiente opción en un programa con interfaz gráfica:

- **ascii** para tipos de archivos ascii.
- **binary** para tipos de archivos binarios.

Establecimiento de permisos

El protocolo FTP sigue los permisos establecidos en entornos de tipo UNIX y sus similares GNU/Linux, con lo cual existen tres grupos de permisos en el siguiente orden: propietario, grupo y otros:

- **Propietario(user=u)**: El creador o el que ha subido el archivo al servidor FTP.
- **Grupo(group=g)**: Se refiere a un grupo de usuarios que posee la propiedad del archivo, al que probablemente pertenece el propietario.
- **Otros(others=o)**: Son el resto de usuarios no propietarios o que no pertenecen al grupo indicado. Son el resto del mundo.

Cada grupo a su vez puede tener tres permisos en el siguiente orden: lectura, escritura y ejecución identificados respectivamente por una 'r', una 'w' y una 'x'. La ausencia de permiso es identificada con el carácter '-'. Cada permiso tiene un equivalente numérico, así: r=4, w=2, x=1 y -=0. Por ejemplo: rw- identifica permiso de lectura y escritura o lo que es lo mismo 4+2+0=6

En un sistema operativo tipo GNU/Linux mediante el comando 'ls -l' puedes ver los permisos asignados a ficheros y directorios, por ejemplo si la salida del anterior comando es:

```
-rw-r--r-- 1 alumno clase 0 jun 20 01:15 prueba1.txt
```

significa que,

- **prueba1.txt** es un fichero ya que **-rw-r--r--** comienza con '-', si fuese un directorio aparecería un 'd'
- **rw-r--r--** identifica los permisos del fichero prueba1.txt, que divididos 3 a 3 representan de izquierda a derecha: propietario, grupo, otros.
- **rw-** identifican los permisos del usuario propietario, en este caso **alumno**. Por lo tanto alumno posee los permisos de **lectura** y **escritura** sobre el fichero prueba1.txt o lo que es lo mismo $4+2+0=6$
- **r--** identifican los permisos del grupo propietario, en este caso **clase**. Por lo tanto clase posee solamente el permiso de **lectura** o lo que es lo mismo $4+0+0=4$
- **r--** identifican los permisos de los **otros** (resto del mundo). Por lo tanto todos los usuarios que no son alumno y aquellos que no pertenecen al grupo clase poseen solamente el permiso de **lectura** o lo que es lo mismo $4+0+0=4$

Por lo tanto los permisos **rw-r- -r- -** equivalen a **644**.

Por otro lado en un sistema GNU/Linux, en principio, no todos los usuarios del sistema tienen acceso por ftp, así existe un fichero **/etc/ftpusers** que contiene una lista de usuarios que no tienen permiso de acceso por FTP. Por razones de seguridad al menos los siguientes usuarios deberían estar listados en este fichero: root, bin, uucp, news. Ten en cuenta que las líneas en blanco y las líneas que comiencen por el carácter '#' serán ignoradas.

Comandos FTP

Comandos FTP	
ABRIR/CERRAR CONEXIÓN	
COMANDO/S Y ARGUMENTOS	EXPLICACIÓN
open servidor	Inicia conexión remota con un servidor ftp.
close / disconnect	Finalizan la sesión ftp sin cerrar la consola ftp.
bye / quit / exit	Terminan la sesión ftp y salen de la consola ftp.
!	Sale a línea de comandos del sistema operativo temporalmente sin cortar la conexión. Para volver, teclea exit en la línea de comandos.
AYUDA	
COMANDO/S Y	EXPLICACIÓN

ARGUMENTOS	
? / help	Muestra una lista de los comandos disponibles.
? comando / help comando	Muestra la información relativa al comando.
TRABAJAR CON DIRECTORIOS	
COMANDO/S Y ARGUMENTOS	EXPLICACIÓN
cd directorio	Cambia de directorio en el servidor remoto.
lcd directorio	Cambiarse de directorio en el equipo local (cliente ftp).
dir directorio / ls directorio	Listan el contenido del directorio remoto actual.
pwd	Muestra el directorio activo en el servidor.
lpwd	Muestra el directorio activo en el equipo local (cliente ftp).
rmdir directorio	Elimina un directorio vacío en el servidor.
mkdir directorio	Crea un directorio en el servidor.Crea un directorio en el servidor.
TRABAJAR CON FICHEROS	
COMANDO/S Y ARGUMENTOS	EXPLICACIÓN
delete archivo	Borrar un archivo en el servidor remoto.
mdelete patrón	Borrar varios archivos según un patrón.
get archivo	Obtiene archivo en el equipo cliente desde el servidor remoto.
mget archivos	Obtiene varios archivos desde el servidor remoto.
put archivo	Envía un archivo al servidor remoto.
mput archivos	Envía varios archivos al servidor remoto.
rename archivo	Cambia el nombre a un archivo en el servidor.
ascii	Para configurar y transferir archivos tipo ascii.
binary	Para configurar y transferir archivos tipo binario.

less archivo	Leer contenido de archivo mediante el comando less.
TRABAJAR CON PERMISOS	
COMANDO/S Y ARGUMENTOS	EXPLICACIÓN
chmod	Cambio de permisos en el servidor remoto.
umask	Configura el sistema de permisos en el lado remoto.

Transferencia de archivos en modo texto

De entre los clientes tipo texto cabe destacar dos: el cliente en modo texto **ftp** y el cliente en modo texto **lftp**. En GNU/Linux Debian se dispone del cliente modo ftp en una instalación básica. Para poder utilizarlo en el sistema simplemente hay que ejecutarlo como comando: el comando ftp.

Vamos a ver, a continuación, el comportamiento del cliente en modo texto ftp en la conexión al servidor ftp ftp.rediris.es:

1. Básicamente la sintaxis es la siguiente:

ftp [-pinegvd] [host [port]]

donde

- **host** identifica el servidor ftp
 - **port** identifica el puerto, por defecto 21, por lo cual si conectas a un servidor ftp configurado en ese puerto no es necesario escribirlo, ya se considera.
2. Al ejecutar el comando se abrirá una consola propia de ftp en la cual puedes introducir comandos ftp para: abrir conexión, moverse por rutas, descargar archivos ...

Es muy típico ejecutarlo con el parámetro host, esto es, con el servidor ftp al cual quieres conectar:

```
# ftp ftp.rediris.es
```

También puedes ejecutar el comando sin parámetros, de esta forma abrirás directamente la consola ftp y deberás actuar con ella a través de los comandos de la misma:

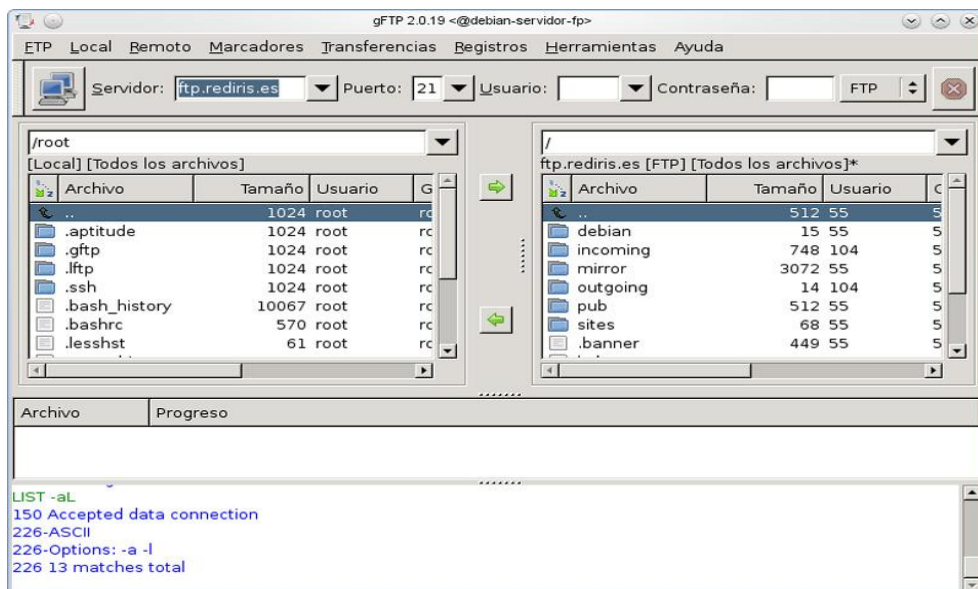
```
# ftp
ftp> o
(to) ftp.rediris.es
```


3. A continuación se pedirá usuario y contraseña para establecer la conexión. En el caso del servidor de rediris puedes conectar mediante un usuario cualquiera y una contraseña cualquiera. Es muy típico en servidores ftp que exista un usuario anónimo, cuya contraseña sea cualquier dirección de correo -real o ficticia-, por ejemplo: a@ .
4. Ahora en la consola ftp puedes ejecutar los comandos que estén habilitados (pueden saberse ejecutando el comando `help`.)

Transferencia de archivos en modo gráfico

Típicamente los clientes gráficos se comportan todos igual, esto es, tienen una interfaz parecida, básicamente presentan una ventana partida en dos secciones: la de la izquierda suele representar el equipo cliente ftp -desde donde se intenta establecer la conexión- y la de la derecha suele representar el equipo servidor ftp -quién recibe la conexión-. Luego suelen existir, en alguna zona determinada de la ventana: en el centro entre las dos secciones, arriba de las dos secciones, etc una serie de botones, usualmente representados como flechas que indican la posibilidad de subir o descargar archivos. Incluso dependiendo del cliente en modo gráfico es posible guardar los datos de las conexiones como plantillas, de tal forma que la próxima vez que intentes establecer la conexión con un mismo servidor ftp en vez de tener que rellenar los campos referentes a la conexión puedes hacerlo a través de la plantilla que ya posee el valor de esos campos.

Dentro de los clientes ftp en modo gráfico cabe destacar dos: **gftp** y **filezilla**. A continuación puedes ver un ejemplo de como utilizarlos para establecer una conexión con un servidor ftp, el servidor ftp.rediris.es :



1. Cliente en modo gráfico **gftp**.
 - **Servidor:** Escribe aquí el nombre o IP del servidor FTP: ftp.rediris.es
 - **Puerto:** Escribe aquí el puerto TCP de la conexión de control, por defecto: 21. Puedes omitirlo siempre y cuando sea el 21.

- **Usuario:** Escribe aquí el usuario con permisos de conexión en el servidor ftp. En la imagen puedes ver que no se ha escrito nada, esto es debido que el servidor ftp.rediris.es permite la entrada a cualquier usuario y el cliente gráfico gftp al intentar conectar te pedirá un usuario que tenga permisos para la conexión. Pulsas en cancelar y gftp cubrirá los campos usuario y contraseña, entrando al servidor ftp.
- **Contraseña:** Escribe aquí la contraseña del usuario con permisos de conexión en el servidor ftp. En la imagen puedes ver que no se ha escrito nada, esto es debido a la misma causa que en el campo Usuario.

2. Cliente en modo gráfico filezilla.

Transferencia de archivos desde navegador

El navegador web también puede ejercer de cliente ftp y, puesto que la mayoría de los sistemas operativos cuentan con un navegador en su instalación, es una de las herramientas más usadas para transferencia de archivos.

Para poder usar el navegador como cliente ftp solamente debes escribir en la barra de dirección una dirección URL tipo, como la siguiente:

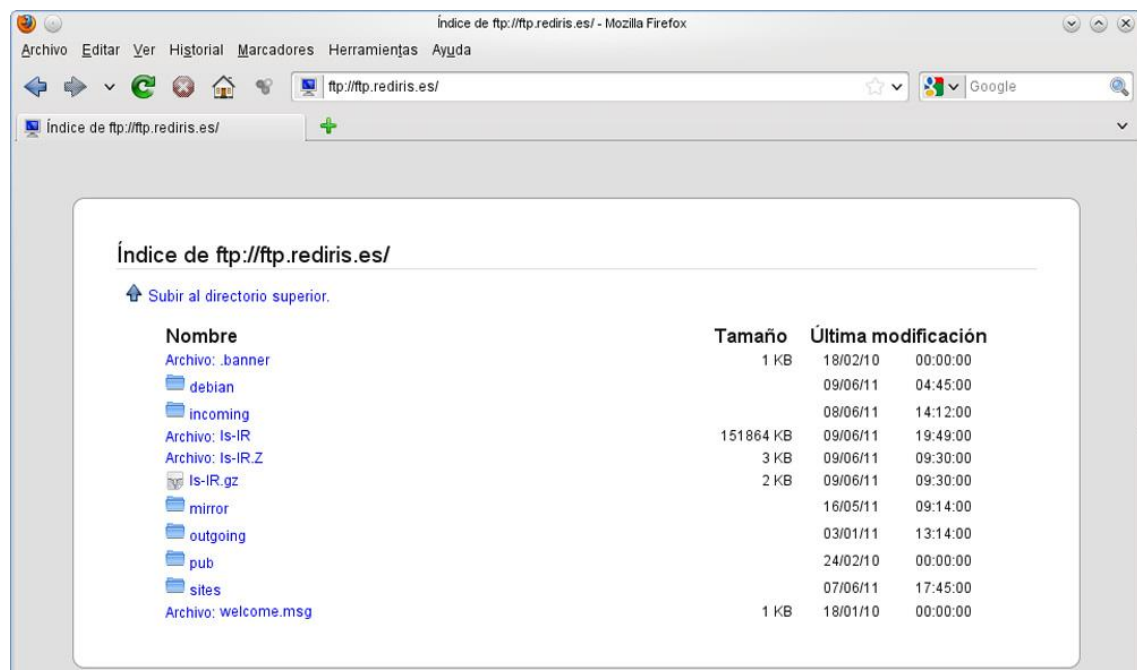
ftp://nombre_servidor_ftp:puerto

donde,

- **ftp://** indica que el protocolo que deseas que interprete el navegador sea el ftp.
- **nombre_servidor_ftp** representa el nombre o la IP del servidor ftp.
- **puerto** indica el puerto TCP, por defecto 21. Puedes omitirlo siempre y cuando sea el 21.

Si el servidor ftp permite la conexión a un usuario anónimo, al ejecutar **ftp://nombre_servidor_ftp:puerto** entrarás directamente al servidor ftp, esto es, el navegador no preguntará qué usuario y contraseña necesitas para establecer la conexión.

En la siguiente imagen puedes ver como puedes acceder al servidor ftp de rediris utilizando el navegador:



Así, lo único que tienes que hacer es escribir en la dirección URL: **ftp://ftp.rediris.es** y pulsar **Enter**, con lo cual, automáticamente, conectas con el servidor ftp, pudiendo visitar las carpetas y ver los ficheros como si de un explorador de archivos se tratará.

Para descargar las carpetas o archivos simplemente debes pulsar con el botón derecho del ratón sobre ellos y elegir la opción **Guardar enlace como...** -que aparece en Firefox y es similar en otros navegadores-.

Pero no todo van a ser ventajas al utilizar el navegador como cliente ftp, puesto que otros clientes tienen la posibilidad de continuar las descargas cuando estás sufrieron algún tipo de interrupción, cosa que no pasa con el cliente ftp del navegador, como por ejemplo el cliente gráfico FileZilla que soporta y reanuda la transferencia de archivos de gran tamaño(> 4 GB).

Asegurar la transferencia de archivos

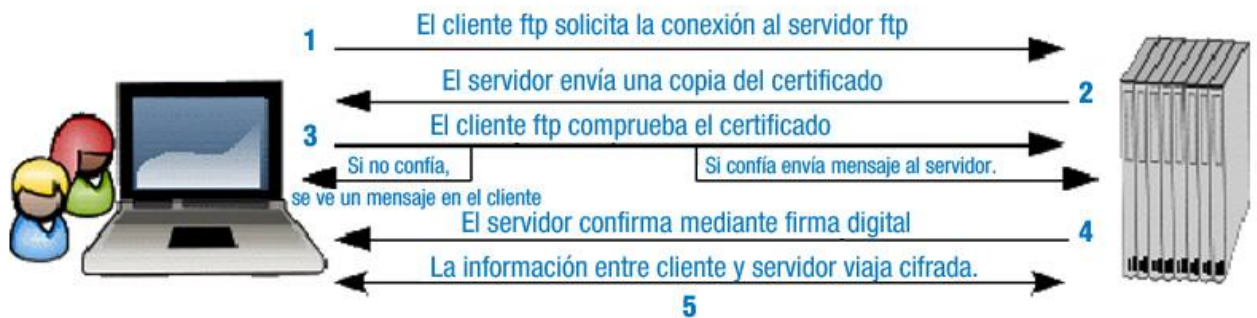
Cuando interese asegurar el servicio de transferencia de archivos debe descartarse el protocolo ftp y empezar a pensar en otras alternativas, como: **ftps** o **sftp**.

FTPS es una extensión del protocolo FTP que asegura el cifrado en la transferencia mediante los protocolos SSL/TLS. Permite tres tipos de funcionamiento:

- SSL Implícito:
 - Como conexiones HTTPS.
 - Usa los puertos 990 y 989.
- SSL Explícito
 - El cliente usa los mismos puertos estándar FTP: 20 y 21 pero se efectúa el cifrado en ellos.
 - Usa AUTH SSL.
- TLS Explícito:
 - Similar a SSL Explícito pero usa AUTH TLS.

El cifrado al que nos referimos es el cifrado de clave pública o asimétrico: **clave pública (kpub)** y **clave privada (kpriv)**. La **kpub** interesa publicarla para que llegue a ser conocida por cualquiera, la **kpriv** no interesa que nadie la posea, solo el propietario de la misma. Ambas son necesarias para que la comunicación sea posible, una sin la otra no tienen sentido, así una información cifrada mediante la **kpub** solamente puede ser descifrada mediante la **kpriv** y una información cifrada mediante la **kpriv** sólo puede ser descifrada mediante la **kpub**.

En el cifrado asimétrico podemos estar hablando de individuos o de máquinas, en nuestro caso hablamos de máquinas y de flujo de información entre el **cliente ftp (A)** y el **servidor ftp (B)**. Ver la siguiente tabla como ejemplo de funcionamiento del cifrado asimétrico:



$A(\text{inf}) \rightarrow \text{inf cifrada} \rightarrow B [\text{descifrar inf}] \rightarrow B(\text{inf}) = A(\text{inf})$

$A(\text{inf}) \rightarrow \text{inf cifrada} = [(\text{inf})]_{k_{\text{pub}}B} \rightarrow B [\text{inf. cifrada}]_{k_{\text{priv}}B} \rightarrow B(\text{inf}) = A(\text{inf})$

Identificación	
A	Cliente ftp.
inf cifrada = $[(\text{inf})]_{k_{\text{pub}}B}$	Información cifrada mediante la clave pública de B obtenida a través de un certificado digital.
$[\text{inf. cifrada}]_{k_{\text{priv}}B}$	Información descifrada mediante la clave privada de B.
B	Servidor ftp.

