

Resumen del tema 4

Despliegue de Aplicaciones Web

14 de enero de 2023

CIFP Carlos III - Cartagena

Santiago Francisco San Pablo Raposo

2º curso DAW

Contenido

**No se encontraron entradas de tabla de contenido.**

# Índice de ilustraciones.

**No se encuentran elementos de tabla de ilustraciones.**

# Índice de tablas.

**No se encuentran elementos de tabla de ilustraciones.**

Resumen del tema 4.

# 1.- Servicio de transferencia de archivos.

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 es el FTP, que utiliza el protocolo FTP empleando la arquitectura cliente-servidor.

El servidor espera peticiones y el cliente, es quien las realiza para poder transferir los archivos al servidor.

**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**. Hoy en día **existen extensiones** sobre el protocolo FTP que aseguran el cifrado en la transferencia, como **FTPS**, **empleando el cifrado SSL/TLS**.

No confundir FTPS con SFTP, ya que este último es implementado con otro servicio, el servicio SSH, y es utilizado para conexiones remotas seguras a través de un terminal de comandos.

**Para saber más**: [Página web del RFC 959 sobre FTP](https://www.rfc-es.org/rfc/rfc0959-es.txt), [Página web del RFC 4251 sobre SSH](https://www.rfc-es.org/rfc/rfc0959-es.txt).

## 1.1.- ¿Cómo funciona?

El protocolo FTP 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.

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.

Otras alternativas al FTP, como el protocolo FTPS sí que mantienen las comunicaciones cifradas. Aún así, FTP está muy extendido en Internet, ya que a menudo los usuarios necesitan transferir archivos entre máquinas sin importar la seguridad.

**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.

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.
* 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.

Texto

Descripción generada automáticamente

## 1.2.- Cliente FTP.

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://

¿Qué cliente elegir?

**Depende**:

* ¿Conoces la consola ftp? **Si te manejas con soltura en la consola ftp puedes pensar en un cliente ftp de comandos** que permita utilizar la tecla "tabulado" después de escribir unos caracteres para complementar los nombres de archivos.
* ¿Cuál es el uso que necesitas? ¿Para qué lo vas a utilizar? **A lo mejor solamente quieres visitar un servidor ftp y descargar un archivo sin tener que andar instalando nuevos programas**. En este caso puedes utilizar el cliente ftp del navegador, ftp://
* **¿Quieres reanudar la conexión en caso de corte en la misma?** En este caso mejor un **cliente tipo gráfico**.
* **¿Deseas facilidad de manejo?** Un cliente terminal de comandos suele ser menos interactivo que uno **gráfico**. En el terminal necesitas saber manejar comandos FTP, mientras que, en un cliente gráfico, puedes manejarte a través de clics de ratón.
* ¿Qué tipo de conexión quieres establecer? ¿cifrada? ¿no cifrada? **Dependiendo del tipo de conexión debes emplear un cliente u otro**, ya que no todos los clientes ftp permiten conexiones cifradas.
* **¿Deseas recordar conexiones (sitios)?** No todos los clientes ftp lo permiten.

**Un cliente FTP muy recomendable es el cliente gráfico FTP FileZilla**, ya que posee las siguientes características:

* Fácil de usar.
* Soporta FTP, FTP sobre SSL / TLS (FTPS) y SFTP.
* Compatibilidad con múltiples plataformas: se ejecuta en Windows, Linux, BSD, Mac OS X y más.
* Soporte Ipv6.
* Disponible en varios idiomas.
* Soporta y reanuda la transferencia de archivos de gran tamaño (mayores de 4 GB).
* Interfaz de usuario con pestañas.
* Potente administrador de sitios y cola de transferencia.
* Marcadores.
* Arrastrar y soltar.
* Permite configurar límites de velocidad de transferencia.
* Nombre de filtros.
* Directorio de comparación.
* Asistente de configuración de la red.
* Edición de archivos remoto.
* Automantenimiento de la conexión.
* HTTP/1.1, SOCKS5 y soporte de FTP-Proxy.
* Fichero de registro.
* Sincronización de directorios de navegación.
* Búsqueda de archivos remoto.

**Para saber más**: [sobre FileZilla, para descargarlo](https://www.youtube.com/watch?v=-AuyBS-N5fQ).

## 1.3.- Tipos de usuarios.

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.
* **Usuarios virtuales (\*)**: es una característica que se ofrece como tercera posibilidad en ciertos servidores como ProFTPD.
  + 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 permiten los siguientes métodos:

* **Ficheros de autenticación del sistema operativo**: /etc/passwd y /etc/shadow: Para ello usa las directivas AuthUserFile y AuthGroupFile.
  + [ProFTPD: Using AuthUserFiles](http://www.proftpd.org/docs/howto/AuthFiles.html)
* **Usuarios virtuales definidos mediante ficheros de autenticación (de texto) propios**, distintos de los del S.O: para ello también usa las directivas **AuthUserFile y AuthGroupFile**.
* **Autenticación PAM**: es necesario la directva AuthPAMAuthorative a ‘on’.
  + [Información sobre PAM](https://aula21-my.sharepoint.com/personal/5634224_alu365_murciaeduca_es/Documents/Documentos/FP/DAW/Curso%202/Despliegue%20de%20aplicaciones%20web/Evaluacion%202/Tema%204/Recursos/pam_doc.pdf).
* **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.
  + [ProFTPD: SQL and mod\_sql](http://www.proftpd.org/docs/howto/SQL.html)
* **LDAP**: para ello emplea el módulo mod\_ldap.
* **RADIUS**: para ello emplea el módulo mod\_radius.

## 1.4.- Modos de conexión del cliente.

El servidor FTP, a diferencia de otros servidores, **necesita dos puertos TCP para hacer posible la transferencia de archivos**. Ahora bien, ¿son estos puertos siempre los mismos o no? ¿**son independientes del tipo de cliente y servidor o no**? Pues, básicamente **depende de dos factores**:

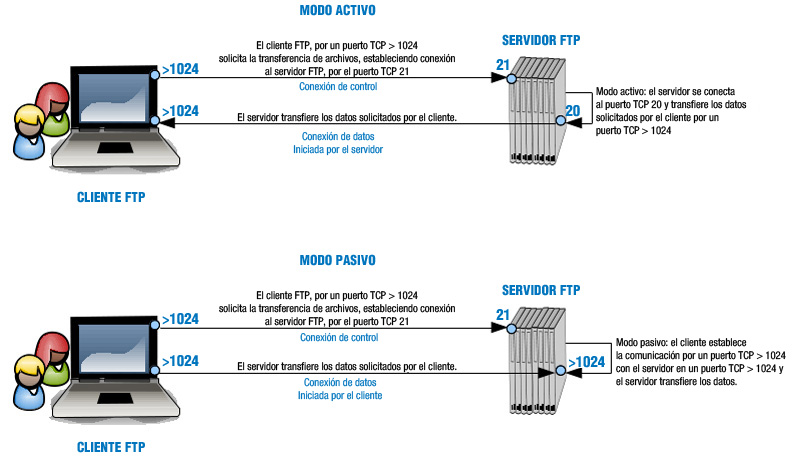
* Del modo de conexión del cliente FTP.
* De la configuración del servidor FTP.

A priori, si no modificamos la configuración del servidor FTP esté otorgará siempre el puerto TCP 21 para el canal de conexión de control. Es **el puerto del canal de transmisión de datos**, el que **varía según 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:

* **Modo activo**: es el método original utilizado por el protocolo FTP.
  + **El servidor abre una conexión desde el puerto 20**, 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**.
* **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 que requiera.



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**.

## 1.5.- Tipos de archivos en la transferencia.

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**: ejecutables (.exe), imágenes (.jpg, .png…), archivos de audio (.mp3, .wav), vídeo (.avi, .mov…), etc.

Es **importante saber el tipo de archivo** con el que estás trabajando en una transferencia, ya que **si no utilizas las opciones adecuadas puedes destruir la información del archivo**.

El servidor FTP permite ambos tipos de archivos, es por eso que al ejecutar el cliente FTP, antes de transferir un arhcdivo, debes utilizar unod e 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.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

## 1.6.- Establecer permisos en FTP.

Existen tres grupos de permisos, en el siguiente orden (al igual que en entornos tipo UNIX y GNU/Linux):

* **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)**: el resto de los usuarios no propietarios o que no pertenecen al grupo indicado. Son el resto del mundo.

Cada uno 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:

* r=4
* w=2
* x=1
* - = 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 comando ‘ls -l’ 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 una ‘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**.

Tipos de permiso de acceso

* **Read (Lectura)**: Permiso para ver el archivo o directorio, sin hacer cambios.
* **Write (Escritura)**: Permiso de escritura: puede escribir el archivo, y, por tanto, cambiarlo.
* **Execute (Ejecución)**: El archivo o directorio puede ser ejecutado. Se usa para directorios con scripts, formularios,…etc.

Tratándose de directorios

* **Read**: Listar los archivos de un directorio.
* **Write**: Añadir nuevos archivos al directorio.
* **Execute**: Acceder a los archivos del directorio.

Uso de chmod con números

Para comprender el significado hay que tener en cuenta que convencionalmente 4 significa permiso de lectura. 2 permiso de escritura y 1 permiso de ejecución. Sumando estos valores vemos que un archivo puede tener los siguientes permisos (por cada tipo de usuario): 4= lectura, 2= escritura y 1= ejecución

* **6 (4+2)**= lectura y escritura
* **5 (4+1)**= lectura y ejecución
* **3 (2+1)**= escritura y ejecución
* **7 (4+2+1)**= lectura, escritura y ejecución

**Por ejemplo**: *chmod 766* significa que el owner tiene permiso de lectura, escritura y ejecución, y el group y others permiso de lectura y escritura.

Uso de chmod con letras

* 0 = **---** = sin acceso
* 1 = **--x** = ejecución
* 2 = **-w-** = escritura
* 3 = **-wx** = escritura y ejecución
* 4 = **r--** = lectura
* 5 = **r-x** = lectura y ejecución
* 6 = **rw-** = lectura y escritura
* 7 = **rwx** = lectura, escritura y ejecución

**Así, en el ejemplo de antes**:  rw-r--r— significa que:

* el owner tiene permisos de lectura y escritura (rw-)
* el group permisos de lectura únicamente (r--)
* y other, permiso de lectura (r--).

¿Cómo sería el equivalente numérico? Sería chmod 644.

**La sintaxis para usar chmod con texto**: chmod [ugo][+-][rwx] [nombre\_archivo]

**Para saber más**: es conveniente que le des un vistazo al manual de **chmod** y **umask**: **man chmod** y **man umask**.

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.

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza media

## 1.7.- Servicio de transferencia de archivos en modo texto.

Los clientes en modo texto desde siempre estuvieron incorporados en las distribuciones GNU/Linux.

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 6 se dispone del cliente modo ftp en una instalación básica.

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

1. **Sintaxis**: ftp [-pinegvd] [host [port]]
   1. **host**: identifica el servidor FTP.
      1. Es muy típico ejecutarlo indicando el host.
   2. **port**: identifica el puerto por defecto 21, por lo cual, si conectas a un servidor FTP configurado en ese puerto, no es necesario escribirlo.

Puedes ver la ayuda del comando ftp mediante: **man ftp** ó **info ftp**.

1. 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, etc.

Aunque es muy típico ejecutar el comando FTP indicando el host, es posible 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:

root@debian-servidor-fp:~# ftp

ftp> o

(to) ftp.rediris.es

1. **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@



**Para saber más**: puedes ver un [ejemplo de conexión a un servidor FTP utliizando lftp](https://aula21-my.sharepoint.com/personal/5634224_alu365_murciaeduca_es/Documents/Documentos/FP/DAW/Curso%202/Despliegue%20de%20aplicaciones%20web/Evaluacion%202/Tema%204/Recursos/Ejemplo%20de%20conexion%20FTP%20mediante%20el%20comando%20lftp.txt) en lugar de ftp.

### 1.7.1.- Comandos FTP.

Algunos de los más empleados son los recogidos en la siguiente tabla:

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 argumentos | Explicación |
| ? / 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. |

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 un 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. |

**Para saber más**: puedes descargar un documento con una tabla con [más comandos FTP](file:///D:\OneDrive\OneDrive%20-%20Consejería%20de%20Educación%20y%20Cultura\Documentos\FP\DAW\Curso%202\Despliegue%20de%20aplicaciones%20web\Evaluacion%202\Tema%204\Recursos\DAW04_tabla_comandos_ftp.pdf).

## 1.8.- Servicio de transferencia de archivos en modo gráfico.

Existe la posibilidad de trabajar de otro modo más interactivo, mediante clientes ftp de modo gráfico o mediante el navegador, ya que éste incorpora su propio cliente ftp.

Típicamente **los clientes gráficos se comportan todos igual**:

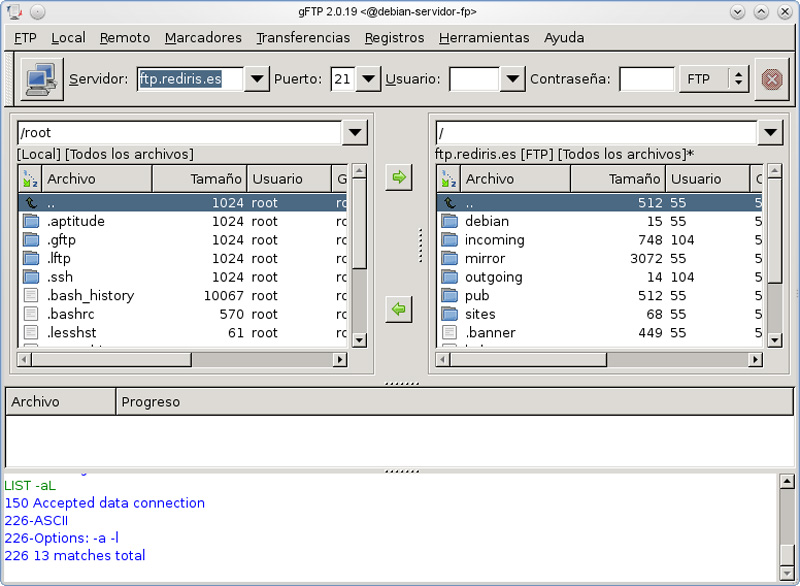
* Tienen una parte de la izquierda que representa el equipo cliente FTP.
* Y la de la derecha suele representar el equipo servidor.

Arriba de estas dos secciones suele haber una serie de botones, usualmente representados como flechas que indican la posibilidad de subir o descargar archivos.

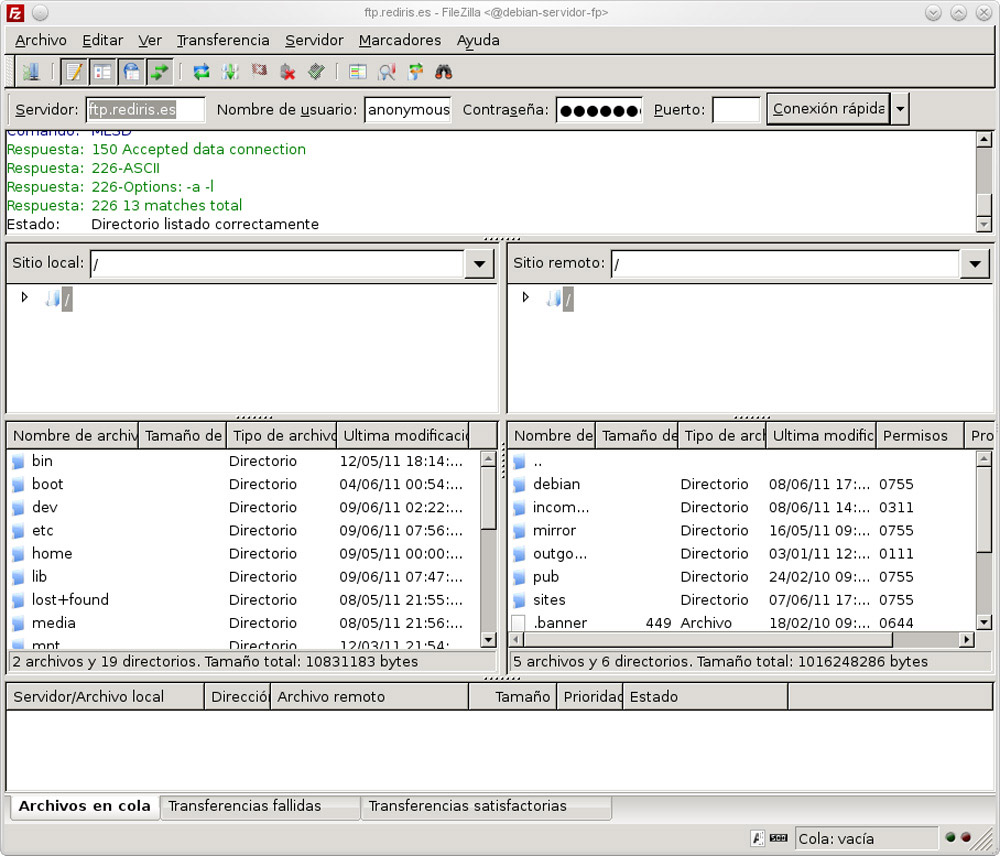
También es posible guardar los datos de las conexiones como plantillas, es decir, que se queden guardadas las credenciales para acceder a un servidor FTP.

A continuación, puedes ver un **ejemplo de conexión** al servidor [ftp.rediris.es](ftp://ftp.rediris.es) desde gftp y otro desde FileZilla.

1. **Cliente en modo gráfico gftp**:
   1. **Servidor**: escribe aquí el nombre o IP del servidor FTP: [ftp.rediris.es](ftp://ftp.rediris.es)
   2. **Puerto**: Escribe aquí el puerto TCP de la conexión de control, por defecto: 21. Puedes omitirlo siempre y cuando sea el 21.
   3. **Usuario**: escribe aquí el usuario con permisos de conexión en el servidor FTP.
      1. 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.
   4. **Contraseña**: escribe aquí la contraseña del usuario con permisos de conexión al servidor FTP.



1. **Cliente en modo gráfico FileZilla**:
   1. **Servidor**: Escribe aquí el nombre o IP del servidor FTP: ftp.rediris.es
   2. **Nombre de usuario**: Escribe aquí el usuario con permisos de conexión en el servidor ftp.
      1. Filezilla, al contrario que gftp, no cubre los datos usuario y contraseña si tú no escribes nada en los campos, entonces debes escribir un nombre de usuario, por ejemplo anonymous, y una contraseña -cualquier secuencia de caracteres-.
   3. **Contraseña**: Escribe aquí la contraseña del usuario con permisos de conexión en el servidor ftp.
   4. **Puerto**: Escribe aquí el puerto TCP de la conexión de control, por defecto: 21. Puedes omitirlo siempre y cuando sea el 21.



**Para saber más**: instalación del cliente gftp:

1) Actualizar sistema operativo GNU/Linux Debian 6 Squeeze:

root@debian-servidor-fp:~# apt-get update

root@debian-servidor-fp:~# apt-get upgrade

2) Instalar cliente ftp en modo gráfico: gftp

root@debian-servidor-fp:~# apt-get install gftp

[Instalación y uso de FileZilla](https://www.youtube.com/watch?v=nBm-2rgkf5Y&t=17s).

## 1.9.- Servicio de transferencia de archivos desde el 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.

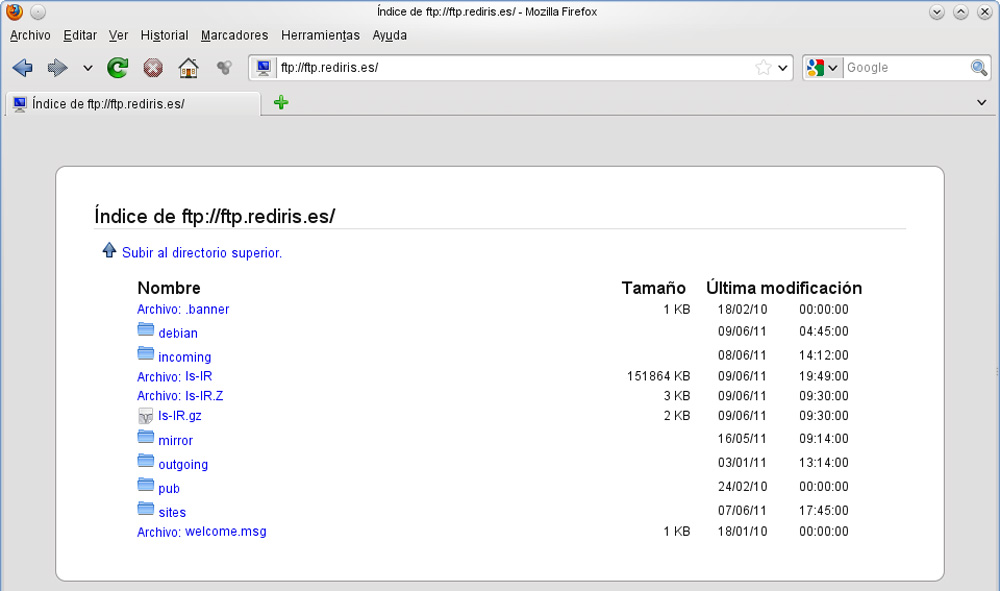
solamente debes escribir en la barra de dirección una dirección URL tipo, como la siguiente:

**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 el 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.



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). Además, tampoco te permite subir archivos o carpetas.

## 1.10.- Asegurando el servicio de transferencia de archivos.

**Empleando** **el protocolo ftp** cualquiera que tenga acceso al canal de transmisión podrá ver en texto claro todo lo que se transmite, esto es, **los datos no se cifran**.

Entonces, **cuando interese asegurar el servicio de transferencia de archivos debes descartar 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:

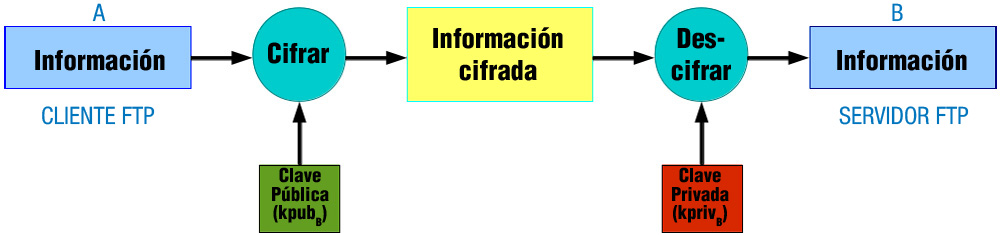
* **SSL Implícito**:
  + Como conexiones HTTPs.
  + Usa los puertos 990 y 989.
* **SSL Explícito**:
  + El cliente usa los mismos puertos estándar FTP: 20 y 21 pero se efectúa el cifrado en ellos.
  + Usa AUTH SSL.
* **TLS Explícito**:
  + Similar a SSL Explícito pero usa AUTH TLS.

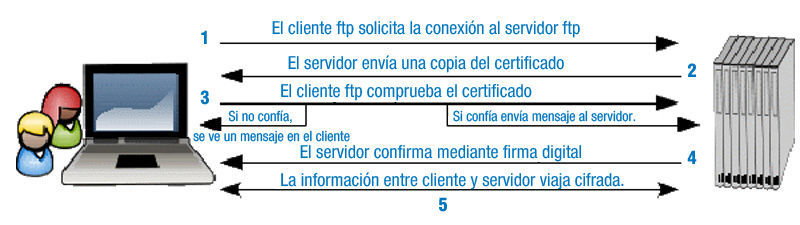
El cifrado al que nos referimos es el [cifrado de clave pública o asimétrico](http://www.criptored.upm.es/intypedia/video.php?id=criptografia-asimetrica&lang=es): **clave pública(kpub)** y **clave privada(kpriv)**.

* 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)**.





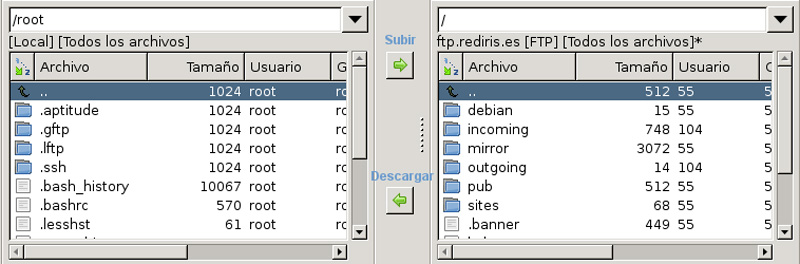
A(inf) → inf cifrada → B [descifrar inf] → B(inf) = A(inf)

A(inf) → inf cifrada = [(inf)]kpubB → B [inf. cifrada]kprivB → B(inf) = A(inf)

* **A**: cliente FTP.
* **Inf cifrada = [(inf)]kpubB**: información cifrada mediante la clave pública de B obtenida a través de un certificado digital.
* **[inf. Cifrada]kprivB**: información descifrada mediante la clave privada de B.
* **B**: servidor FTP.

## 1.11.- El servicio de transferencia de archivos en el proceso de despliegue de una aplicación web.

**Reflexiona**: ¿Cómo actualizas una página web de forma remota? ¿Qué servicios suelen ofrecer las empresas de alojamiento web para que puedas subir archivos a tus aplicaciones? ¿Cuánto tiempo se mantiene la conexión subiendo un archivo?



**Suele ser típico que cualquier aplicación web en Internet disponga de la posibilidad de subir archivos** mediante una configuración del código fuente de la misma, una aplicación propia o una aplicación de terceros, como los **paneles de administración web**.

Si empleas una aplicación web para subir archivos debes **tener en cuenta cuánto tiempo puedes mantener la conexión abierta con el servicio web y cuál es el tamaño máximo de subida de un archivo**. Estas cuestiones suelen ser típicas de la configuración del servidor web. Por la contra, si empleas un servidor ftp dependerá de éste las cuestiones anteriores.

**Se suele configurar el servidor web con unos parámetros**: tiempo de conexión y tamaño máximo de subidas de archivos diferentes del servidor ftp. De tal forma que, para archivos de tamaño no muy grandes, se puedan emplear aplicaciones web y no se sufra un corte en la subida de archivos, y para archivos grandes, se emplee el servidor FTP.

Normalmente **las empresas de alojamiento web** (hosting) permiten la subida de archivos mediante un servidor ftp y **poseen documentación que explica cómo conectarse a sus servidores ftp** a través de algún cliente ftp, como por ejemplo: FileZilla, Cute FTP, Fetch o Transmit. También **suelen permitir usar SCP o SFTP** para transferir ficheros de forma segura mediante un canal cifrado.

Por ejemplo, **en FileZilla se puede establecer la conexión de forma cifrada directamente**, indicando como puerto TCP **el número del servidor SSH, por defecto, 22**.

**También debes saber que muchos editores web permiten subir tu aplicación web al servidor con el protocolo FTP**, esto te puede resultar más sencillo que el uso de una aplicación de FTP independiente. Eso si, **sea cual sea el método ftp que utilices para subir archivos y actualizar tu web, se desaconseja el uso de aplicaciones no actualizadas** que podrían comprometer la seguridad de tu web.

**A continuación, puedes ver errores típicos**, junto con sus soluciones, que puedes encontrar al subir tu aplicación mediante un servidor FTP:

* Tu cliente FTP muestra el error **access denied**, o similar, cuando subes o borras ficheros y carpetas: **Comprueba que tu usuario FTP tenga permisos suficientes** sobre la carpeta o fichero en la que deseas subir o que deseas borrar.
* **Tus páginas no son reconocidas de forma automática al acceder a tu dominio**: Los servidores GNU/Linux **son sensibles a mayúsculas y minúsculas** por lo que **verifica el nombre de tus archivos**.
* El cliente de FTP te muestra el mensaje **too many connections from your IP address**: Esto quiere decir que existen más conexiones abiertas con el servidor FTP desde la misma dirección IP de las permitidas.
  + Asegúrate que no exista ninguna aplicación, como un cortafuegos, que pueda estar bloqueando las conexiones abiertas, y provocando, de esta forma, que se establezcan más intentos de conexión de los necesarios.

# 2.- Instalación del servidor proftpd.

**Caso práctico**: tendremos que tener en cuenta una serie de cuestiones:

1. **Configurar un servidor FTP** para que en la subida de archivos la conexión no se corte, que es lo más probable que ocurra a través de la aplicación web.
2. Mirar en el servicio contratado el **nivel de cuota en disco**.
3. **Crear un usuario virtual para que pueda subir archivos**. Opciones:
   1. Mediante BB.DD SQL.
   2. Mediante un archivo de autentificación: más útil si es solo para crear un solo usuario.
4. **Si se necesita crear otro usuario, pues otro en el archivo de autenticación** y listo. ¿Y si necesito crear grupos? Pues lo mismo, en un archivo de autenticación de grupos.
5. **Empleo del protocolo SSL o TLS** (TLS es el sucesor de SSL) para cifrar los datos.
   1. Pero si empleo cifrado, voy a tener que utilizar otros puertos TCP distintos del servidor FTP que maneja por defecto: el 21 para la conexión de control y el 20 para la conexión de datos.
6. **Mejor TLS Explícito**, de forma que pueda seguir utilizando los mismos puertos 20 y 21 para el cifrado.
7. Y todo esto **no debe modificar la configuración ya realizada para otras empresas**.

**Todo esto se aborda con un servidor FTP**: el servidor ProFTPD es un buen ejemplo.

1. Comprobar si en este servidor dedicado está instalado.
2. Configurar el servidor proftpd de la siguiente forma:
   1. Configuración independiente para esta empresa.
   2. Usuario virtual en un archivo de autenticación.
   3. Cifrado TLS Explícito para asegurar el cifrado.
   4. Cuota de disco.
   5. Permisos de subida en la carpeta upload correspondiente.

**ProFTPd** es un servidor FTP bajo licencia GPL **altamente configurable**. Permite:

* **Usuarios virtuales con**:
  + LDAP.
  + BB.DD: MySQL, PostgreSQL.
  + Ficheros de autenticación (ficheros de texto).
* **Personalizar opciones según usuario/grupo**.
* Seguridad mediante **cifrado SSL/TLS**.
* Configuraciones independientes mediante **virtualhosts**.

Para instalarlo:

1. **sudo apt install proftpd**

En la instalación deberás elegir si ProFTPd va a ejecutarse como un servicio desde **inetd** o como un **servidor independiente**. Ambas opciones tienen sus ventajas:

* Si solo recibes unas pocas conexiones FTP diarias, probablemente sea mejor ejecutar ProFTPd desde inetd para ahorrar recursos.
* Con más tráfico, ProFTP debería ejecutarse como un servidor independiente para evitar crear un proceso nuevo por cada conexión entrante.

En la instalación se crearán los usuarios **proftpd** y **ftp** con grupo **nogroup** y sin posibilidad de acceso a una consola del sistema. **Se puede comprobar en el fichero /etc/passwd**, donde encontrarás nuevas líneas similares a las siguientes:

proftpd:x:106:65534::/var/run/proftpd:/bin/false

ftp:x:107:65534::/home/ftp:/bin/false

**Para saber más**: [Vídeo sobre instalación y uso de ProFTPd](https://www.youtube.com/watch?v=ijoI1lTDpcI).

## 2.1.- Configuración de proftpd.

Su configuración es similar a la del Servidor Apache:

* **Tiene un fichero de configuración principal**: /etc/proftpd/proftpd.conf
* **La posibilidad de configurar host virtuales o virtualhost**, con sus correspondientes dominios, y todo lo que no esté incluido en la configuración de cada Virtual Host, se heredará de la configuración principal.
* **Configuración a través de directivas**.
* **Contextos de configuración**: global, directorio, virtualhost, Anonymous.
* **Modularización**: al igual que Apache, se pueden activar/desactivar funcionalidades a través de módulos.

**Debes conocer**: [página web oficial de ProFTPd](http://www.proftpd.org/). Información sobre las directivas de ProFTPd.

Una vez instalado ProFTPD existirán **dos ficheros de especial interés**:

* **El fichero /etc/ftpusers**: contiene una lista de usuarios que no tienen permiso de acceso por FTP. Esto es por razones de seguridad.
  + **Los usuarios que se recomiendan listar en este archivo para que no tengan permisos son**: root, bin, uucp, news.
  + **Las líneas en blanco y líneas que comiencen por ‘#’ serán ignoradas**.
* **Fichero de configuración principal (/etc/proftpd/proftpd.conf)**: [similar a este](https://aula21-my.sharepoint.com/personal/5634224_alu365_murciaeduca_es/Documents/Documentos/FP/DAW/Curso%202/Despliegue%20de%20aplicaciones%20web/Evaluacion%202/Tema%204/Recursos/Archivos%20proftpd%20plataforma/DAW04_proftpd.conf).

En el fichero proftpd.conf

* Las líneas que comienzan por **Include** recogerán la configuración de los ficheros que la acompañan.
* **User proftpd** y **Group nogroup** identifican al usuario y grupo con el que se ejecuta proftpd.
* **El soporte LDAP, SQL, TLS, virtualhosts y cuotas están desactivados**. Ver líneas:

#Include /etc/proftpd/ldap.conf

#Include /etc/proftpd/sql.conf

#Include /etc/proftpd/tls.conf

#Include /etc/proftpd/virtuals.conf

QuotaEngine off

* El **mensaje de bienvenida** se encuentra en el fichero **welcome.msg**.
* **Está configurado por defecto el modo de conexión FTP activo** en el puerto TCP 21.
* **Los usuarios que puedan conectarse por FTP**:
  + **Necesitan una consola de comandos activa**, esto es, debe poseer una consola presente dentro del fichero **/etc/shells**.
  + **Pueden moverse por todo el sistema de ficheros**, esto es, no están encerrados (jaula **chroot**) en sus directorios /home, puesto que la directiva **DefaultRoot ~** está comentada.
    - **Por seguridad sería conveniente descomentar la línea** y recargar la configuración del servidor.
* **Para evitar ataques de denegación de servicio** solamente se permiten 30 conexiones simultáneas: **MaxInstances 30**.
* Los permisos para los ficheros y directorios creados en la conexión FTP son: 644 y 755, respectivamente. Ya que **umask 022 022**, donde:
  + El **primer grupo** de tres números identifican los permisos de los **ficheros**.
  + El **segundo grupo** identifica los permisos de los **directorios**.
* Encontrarás al final del mismo un ejemplo de configuración para usuarios anónimos.

Una vez retocada la configuración del servidor ProFTPd, solo reconocerá estos cambios cuando recargues su configuración, con lo cual debes ejecutar el comando:

/etc/init.d/proftpd restart

Si la configuración es correcta, y no quieres reiniciar proftpd, **puedes recargar la configuración mediante el comando**:

/etc/init.d/proftpd reload

**Caso práctico**: archivo de configuración comentado. [Ejemplo proftpd.conf](https://aula21-my.sharepoint.com/personal/5634224_alu365_murciaeduca_es/Documents/Documentos/FP/DAW/Curso%202/Despliegue%20de%20aplicaciones%20web/Evaluacion%202/Tema%204/Recursos/Archivos%20proftpd%20plataforma/proftpd.conf-plantilla.txt).

**NOTA**: No está incluida la parte de configuración de un servidor anónimo, que aparece aunque de forma comentada en el fichero por defecto que nos proporciona la configuración por defecto como hemos visto anteriormente.

## 2.2.- Configurar el servidor como FTP privado.

Tal y como viene el fichero proftpd.conf configurado, por defecto, **podríamos iniciar sesión en el servidor FTP con cualquier usuario del sistema** que posea una consola de comandos activa definida en /etc/shells. Es decir, **por defecto, ya es privado**.

**Por ejemplo**: ftp usuario\_del\_sistema@servidor\_ftp, donde:

**servidor\_ftp:** puede ser el nombre de tu servidor ftp en /etc/hosts o resuelto por DNS o la IP de tu servidor ftp.

**Para saber más**: [ejemplos de configuraciones del servidor ProFTPD](http://www.proftpd.org/docs/example-conf.html).

## 2.3.- Configurar el servidor como FTP privado y anónimo.

**Igual te interesa contar en tu servidor FTP con un usuario anónimo**, el cual establecerá la conexión con cualquier contraseña, y tendrá permisos diferentes a los usuarios del sistema (privados).

**Normalmente los permisos del usuario anónimo en un servidor ftp se establecerán para que solamente pueda moverse por los directorios y descargar archivos**, nunca subirlos, esto es, normalmente el usuario anónimo no podrá crear ni eliminar ficheros y directorios.

Hay que modificar, pues, el fichero proftpd.conf. Al final del archivo, incorpora las siguientes líneas:

# Inicio de la configuración Anonymous

# Usuario anónimo que entrará en el directorio ~ftp, esto es, en la variable $HOME del usuario ftp

# En Debian 6 (squeeze) ~ftp=/home/ftp . Este directorio será la raíz de los directorios en la conexión

# ftp, esto es, /home/ftp estará enjaulado (chroot) de tal forma que aunque el usuario anónimo ftp

# quisiera acceder a otros directorios situados fuera de /home/ftp no podrá acceder.

<Anonymous ~ftp>

# Después de hacer login anónimo mediante ftp el servidor se ejecuta con el usuario ftp y con el grupo

# nogroup

User ftp

Group nogroup

# La siguiente línea permite hacer login con el usuario “anonymous” igual que si fuera el usuario “ftp”

UserAlias anonymous ftp

# Cambios de apariencia, todos los ficheros parecerán pertenecer al usuario y grupo ftp

DirFakeUser on ftp

DirFakeGroup on ftp

# No es necesario tener una shell en /etc/shells

RequireValidShell off

# Limitar el máximo número de logins anónimos concurrentes a 10

MaxClients 10

# Mensaje de conexión en el fichero welcome.msg

DisplayLogin welcome.msg

# No permitir ESCRITURA en cualquier directorio al usuario anonymous, alias del usuario ftp

<Directory \*>

<Limit WRITE>

DenyAll

</Limit>

</Directory>

# Fin de la configuración Anonymous

</Anonymous>

Ahora simplemente, reiniciamos el servidor o recargamos la configuración para que los cambios surtan efecto:

/etc/init.d/proftpd restart ó /etc/init.d/proftpd reload.

Esta configuración permitirá conectar al servidor ftp mediante el usuario **ftp** o por su alias **anonymous** empleando una contraseña cualquiera. Una vez conectado solamente tendrá acceso al contenido de la carpeta /home/ftp y no podrá subir ni eliminar nada de ella.

## 2.4.- Configurar el servidor como FTP anónimo.

Puedes configurar ProFTPd para que solo permita conexiones de usuarios anónimos, es decir, obligando a conectar sin poseer una contraseña del sistema (entrando con una contraseña cualquiera).

**Cambia la configuración del usuario anonymous** ftp de tal forma que al final del fichero aparezcan las siguientes líneas:

<Anonymous ~ftp>

User ftp

Group nobody

# No es necesario tener una shell en /etc/shells

RequireValidShell off

# No se requiere contraseña en la conexión

AnonRequirePassword off

# No permitir ESCRITURA en cualquier directorio al usuario ftp

<Directory \*>

<Limit WRITE>

DenyAll

</Limit>

</Directory>

</Anonymous>

**Puedes crear un usuario anónimo con carácter privado**, esto es, que requiera contraseña para establecer la conexión.

<Anonymous ~invitado>

User invitado

Group nobody

# Se requiere la contraseña de sistema del usuario invitado en la conexión

AnonRequirePassword on

# No permitir ESCRITURA en cualquier directorio al usuario invitado a no ser que establezca conexión

# de la red 192.168.200.

<Directory \*>

<Limit WRITE>

Order allow, deny

Allow from 192.168.200.

Deny from all

</Limit>

</Directory>

</Anonymous>

**Puedes convertir cualquier usuario privado** (del sistema) **que posea una consola de comandos válida en /etc/shell en un usuario anónimo**. **Por ejemplo**: en las configuraciones anteriores sólo tendrías que sustituir el usuario 'ftp' y el usuario 'invitado' por el nombre de un usuario existente en el sistema operativo.

**Debes conocer**: [directivas Order, Allow y Deny de ProFTPd](http://www.proftpd.org/docs/howto/Limit.html).

**Ejercicio resuelto**: Según lo visto anteriormente, ¿cómo permitirías al usuario invitado establecer conexión desde las redes: 192.168.200 y 10.0.200?

Añadiendo en la misma línea de la directiva Allow from 192.168.200. la nueva red separando las redes mediante una coma, tal que así: **Allow from 192.168.200.,10.0.200**.

**Ejercicio resuelto**: Y si además, ¿quisieras permitir el acceso desde los dominios tuhostA.tudominio.edu, tuhostB.tudominio.edu y tuhostC.tudominio.edu?

Pues, **de la misma forma que anteriormente**, añadiendo en la misma línea de la directiva **Allow from 192.168.200** las nuevas redes o dominios separándolos mediante signos coma, tal que así:

**Allow from 192.168.200., 10.0.200.,tuhostA.tudominio.edu,tuhostB.tudominio.edu,tuhostC.tudominio.edu**

## 2.5.- Configurar el servidor FTP con múltiples dominios.

Esto se puede hacer mediante la configuración de host virtuales o virtualhosts. **Estos permiten que un mismo servidor FTP pueda alojar múltiples dominios**. De esta forma, por ejemplo, cada empresa podría tener su virtualhost único e independiente de los demás.

Una cosa que hay que tener en cuenta es que **todo aquello que no esté incluido en la definición de cada virtualhost se heredará de la configuración principal** (proftpd.conf). **Por ejemplo**: en proftpd.conf puedes encontrar la directiva **TimeoutIdle 1200**, que establece la directiva **TimeoutIdle** igual a 1200 segundos, esto es, indica el número máximo de segundos que puede estar un usuario sin hacer nada, pasado ese tiempo se cierra la conexión ftp. Esta es una directiva que interesa tenerla puesta en todos los virtualhost.

**En la definición de la directiva VirtualHost podemos poner la IP del servidor FTP ó bien el nombre DNS correspondiente**. En nuestro escenario, la IP\_Servidor\_FTP=192.168.200.250, **ftp.empresa1.com** y **ftp.empresa2.com** identifican a la misma máquina.

Hay que tener en cuenta que, si las IP empleadas son **IP privadas**, sin existencia en Internet, siempre que se haga referencia a las mismas a través de nombre de dominios, deberá existir un **servidor DNS** que las resuelva en local o bien, en su defecto, deberán existir las entradas correspondientes en el fichero del sistema local **/etc/hosts**.

Independientemente de si configuras virtualhosts basados en IP o en nombre, puedes utilizar usuarios del sistema, pero **también puedes crear los usuarios virtuales que quieras en un fichero similar a /etc/passwd** y llamarlo en la configuración mediante la directiva AuthUserFile, entonces, **para nuestro ejemplo**:

* **Ejecuta el siguiente comando que creará un fichero de autentificación para usuarios virtuales**.

ftpasswd --passwd --name user-empresa1 --file /etc/passwd.usuarios.virtuales --uid 107 --home /var/ftp/empresa1 --shell /bin/false

**Donde**:

* + **ftppasswd**, es el comando que permite crear los usuarios virtuales.
  + **- -passwd**, es el parámetro que pedirá la contraseña del usuario.
  + **- -name user-empresa1**, identifica al usuario virtual de nombre user-empresa1.
  + **- -file /etc/passwd.usuarios.virtuales**, creará, en caso de no existir, o modificará, en caso de existir el fichero de autenticación de usuarios virtuales.
  + **- -uid 107**, es el identificador perteneciente al usuario del sistema **ftp**. Se puede saber ejecutando el comando: **id ftp** .
  + **- -home /var/ftp/empresa1**, identifica a donde se conecta el usuario.
  + **- -shell /bin/false**, identifica una consola de comandos que no permite conexión como usuario del sistema.
* **Ejecuta también el comando:**

ftpasswd --passwd --name user-empresa2 --file /etc/passwd.usuarios.virtuales --uid 107 --home /var/ftp/empresa2 --shell /bin/false

A continuación prosigues, dependiendo si deseas configurar virtualhosts basados en IP o virtualhosts basados en nombre.

## 2.6.- Virtualhosts basados en nombre.

En la definición de la directiva **VirtualHost**, podemos poner la IP del servidor FTP, o bien el nombre DNS correspondiente. En nuestro escenario, la IP\_Servidor\_FTP=192.168.200.250, [**ftp.empresa1.com**](ftp://ftp.empresa1.com) y [**ftp.empresa2.com**](ftp://ftp.empresa2.com) identifican a la misma máquina y a la misma IP.

Ahora si, **cada virtualhost, así como el servidor principal, deben servir en un puerto TCP distinto**.

Procedimiento

1. En la configuración de ProFTPd (/etc/proftpd/proftpd.conf) debes **activar la configuración del fichero virtuals.conf** descomentando la línea.

Include /etc/proftpd/virtuals.conf

1. **Agrega la configuración virtualhost** para empresa1 en el fichero /etc/proftpd/virtuals.conf

<VirtualHost 192.168.200.250>

Port 2121

ServerName "Servidor FTP empresa1"

AuthUserFile /etc/passwd.usuarios.virtuales

DefaultRoot /var/ftp/empresa1/

RequireValidShell off

</VirtualHost>

1. Agrega la configuración virtualhost para empresa2 en el fichero **/etc/proftpd/virtuals.conf**.

<VirtualHost ftp.empresa2.com>

Port 2122

AuthUserFile /etc/passwd.usuarios.virtuales

ServerName "Servidor FTP empresa2"

DefaultRoot /var/ftp/empresa2/

RequireValidShell off

</VirtualHost>

1. **Configura permisos en las carpetas** /var/ftp/empresa1 y /var/ftp/empresa2 **para los usuarios virtuales**:

chown ftp /var/ftp/empresa1/ /var/ftp/empresa2/

1. **Recarga la configuración del servidor**.

/etc/init.d/proftpd restart

**Explicación fichero VirtualHost**:

* **<VirtualHost IP\_Servidor\_FTP>**: inicio etiqueta **virtualhost**: define la IP del servidor FTP. También puede ser **<VirtualHost Nombre\_DNS\_Servidor\_FTP>**
* **Port numero**: identifica el puerto TCP por el que se espera la conexión al servidor FTP.
* **ServerName “Servidor FTP empresaX”**: Configura el nombre que se muestra en la conexión de los usuarios.
* **DefaultRoot /var/ftp/empresaX/**: Definición de la ruta que sirve ProFTPFD, en este caso: /var/ftp/empresaX/ mediante la directiva DefaultRoot.
  + De esta forma, los usuarios, cuando conecten con el servidor FTP estarán enjaulados en la ruta /var/ftp/empresaX, con lo cual, no podrán acceder a otro directorio que no esté contenido dentro de este.
* **RequireValidShell off**: no es necesario tener una Shell declarada en el fichero /etc/shells.
* **<VirtualHost>**: Fin de la etiqueta **VirtualHost**: fin de la definición de este virtualhost para la empresa1.

## 2.7.- Virtualhosts basados en IP.

En la definición de la directiva **VirtualHost** podemos poner la IP del servidor FTP o bien el nombre DNS correspondiente En nuestro escenario, la IP\_Servidor\_FTP= 192.168.200.250, **ftp.empresa1.com** y **ftp.empresa2.com** identifican a la misma máquina y a distintas IP. **Ahora es indiferente el puerto TCP en el que sirve cada virtualhost**, así como el servidor principal, esto es, puede ser el mismo o no ya que **ahora cada puerto está relacionado con una IP distinta**.

**Este método no aporta ventajas sobre el anterior**, es más, aún **puede ser más difícil de mantener si las IP del servidor FTP se modifican con cierta frecuencia**.

1. Modifica la configuración virtualhost para empresa1 en el fichero **/etc/proftpd/virtuals.conf**.

<VirtualHost 192.168.200.251>

ServerName "Servidor FTP empresa1"

AuthUserFile /etc/passwd.usuarios.virtuales

DefaultRoot /var/ftp/empresa1/

RequireValidShell off

</VirtualHost>

1. Agrega la configuración virtualhost para empresa2 en el fichero **/etc/proftpd/virtuals.conf**.

<VirtualHost ftp.empresa2.com>

AuthUserFile /etc/passwd.usuarios.virtuales

ServerName "Servidor FTP empresa2"

DefaultRoot /var/ftp/empresa2/

RequireValidShell off

</VirtualHost>

1. **Configura permisos en las carpetas /var/ftp/empresa1 y /var/ftp/empresa2 para los usuarios virtuales**:

chown ftp /var/ftp/empresa1/ /var/ftp/empresa2/

1. **Recarga la configuración del servidor**.

/etc/init.d/proftpd restart

Explicación fichero virtualhost:

* **<VirtualHost IP\_Servidor\_FTP>** → Inicio etiqueta virtualhost: define la IP del servidor ftp. También puede ser **<VirtualHost Nombre\_DNS\_Servidor\_FTP>**
* **ServerName "Servidor FTP empresaX"**→ Configura el nombre que se muestra en la conexión de los usuarios.
* **DefaultRoot /var/ftp/empresaX/** → Definición de la ruta que sirve ProFTPFD, en este caso: /var/ftp/empresaX/ mediante la directiva **DefaultRoot**, esto es, indica que los usuarios cuando conecten con el servidor ftp, estarán enjaulados en la ruta /var/ftp/empresaX/, con lo cual no podrán acceder a otro directorio que no esté contenido dentro de éste.
* **RequireValidShell off** → No es necesario tener una shell declarada en el fichero /etc/shells .
* **</VirtualHost>**→ Fin de la etiqueta **VirtualHost**: fin de la definición de este virtualhost para la empresa1.

## 2.8.- Cuotas de disco para los usuarios (I).

El archivo **/etc/proftpd/proftpd.conf** llama mediante la directiva **Include** al archivo **/etc/proftpd/modules.conf** en el que están activadas las cuotas (**LoadModule mod\_quotatab.c, LoadModule mod\_quotatab\_file.c**), luego para activarlas tienes que sustituir en el archivo **/etc/proftpd/proftpd.conf** el código:

<IfModule mod\_quotatab.c>

QuotaEngine off

</IfModule>

Por el siguiente código:

<IfModule mod\_quotatab.c>

QuotaEngine on

QuotaLog /var/log/proftpd/quota.log

<IfModule mod\_quotatab\_file.c>

QuotaLimitTable file:/etc/proftpd/ftpquota.limittab

QuotaTallyTable file:/etc/proftpd/ftpquota.tallytab

</IfModule>

</IfModule>

Donde:

* **<IfModule mod\_quotatab.c> … </IfModule>**→ Indica que si el módulo mod\_quotatab.c está cargado en el archivo /etc/proftpd/modules.conf se realizarán las directivas que contengan.
* **QuotaEngine on** → Activa las cuotas.
* **QuotaLog /var/log/proftpd/quota.log** → Indica el archivo de registro sobre cuotas.
* **<IfModule mod\_quotatab\_file.c> … </IfModule>** → Indica que si el módulo mod\_quotatab\_file.c está cargado en el archivo /etc/proftpd/modules.conf se realizarán las directivas que contengan.
* **QuotaLimitTable file:/etc/proftpd/ftpquota.limittab** → Indica el archivo sobre el límite de cuotas **Limit**.
* **QuotaTallyTable file:/etc/proftpd/ftpquota.tallytab** → Indica el archivo sobre el límite de cuotas **Tally**.

Para ProFTPd, existen **dos tipos de cuotas**:

* **Limit**: Es la cuota que te interesa si estás pensando en restringir el espacio en disco a los usuarios. Éste puede ser **soft**, cuando existe un espacio de gracia(tamaño en bytes) que puede sobrepasar el límite, o **hard** cuando no existe un espacio de gracia.
* **Tally**: Utilizado cuando quieres limitar el número de ficheros que se utilizan.

**Para saber más**: sobre las cuotas, puedes visitar la [documentación oficial de ProFTPd sobre mod\_quotab](http://www.proftpd.org/docs/contrib/mod_quotatab.html).

**Forma más sencilla de crear cuotas**: hacer el símil upload = espacio en disco. Es decir, los archivos subidos no pueden ocupar más del que queramos darle como espacio en disco, ya que los bytes subidos funcionan como espacio en disco, puesto que los bytes cargados a través de FTP se almacenan automáticamente en el disco.

**Crear las cuotas mediante el comando [ftpquota](http://www.proftpd.org/docs/contrib/ftpquota.html" \o "Acceder a la documentación oficial de ProFTPD sobre el comando ftpquota.)**:

# ftpquota --create-table --type=limit --table-path=/etc/proftpd/ftpquota.limittab

# ftpquota --create-table --type=tally --table-path=/etc/proftpd/ftpquota.tallytab

**Por ejemplo, si quisieras limitar a un usuari de nombre user-empresa1 el espacio de subida en 4 GB**:

# ftpquota --add-record --type=limit --name=user-empresa1 --quota-type=user \

--bytes-upload=4 --units=Gb --table-path=/etc/proftpd/ftpquota.limittab

**Y si quisieras limitar la subida y la bajada a 4 y 2 GB respectivamente al usuario user-empresa2**:

# ftpquota --add-record --type=limit --name=user-empresa2 --quota-type=user \

--bytes-upload=4 --bytes-download=2 --units=Gb --table-path=/etc/proftpd/ftpquota.limittab

### 2.8.1.- Cuotas de disco para los usuarios (II).

¿Cómo verificar el funcionamiento de las cuotas?. Y si quisieras comprobar la cuota de un usuario, ¿es posible? ¿Y si quisieras actualizarla? ¿Y desactivarlas para algún usuario? ¿Y borrarlas?

Utilizas el comando **ftpquota**, como sigue:

* **Para ver los registros de cuotas**, esto es, a quién se le está ejerciendo las cuotas:

# ftpquota --show-records --type=limit --table-path=/etc/proftpd/ftpquota.limittab

-------------------------------------------

Name: user-empresa1

Quota Type: User

Per Session: False

Limit Type: Hard

Uploaded bytes: 4294967296.00

Downloaded bytes: unlimited

Transferred bytes: unlimited

Uploaded files: unlimited

Downloaded files: unlimited

Transferred files: unlimited

* **Para actualizar la cuota de un usuario, por ejemplo, user-empresa1**:

# ftpquota --update-record --type=limit --name=user-empresa1 --quota-type=user \

--bytes-upload=2300 --units=Mb --table-path=/etc/proftpd/ftpquota.limittab

**Con lo cual, si compruebas de nuevo los registros, verás que los cambios surgieron efecto**:

# ftpquota --show-records --type=limit --table-path=/etc/proftpd/ftpquota.limittab

-------------------------------------------

Name: user-empresa1

Quota Type: User

Per Session: False

Limit Type: Hard

Uploaded bytes: 2411724800.00

Downloaded bytes: unlimited

Transferred bytes: unlimited

Uploaded files: unlimited

Downloaded files: unlimited

Transferred files: unlimited

**Para desactivar la cuota de un usuario debes borrar el registro**, por ejemplo, user-empresa1:

# ftpquota --delete-record --type=limit --name=user-empresa1 --quota-type=user

Con lo cual, **si compruebas de nuevo los registros, verás que los cambios surgieron efecto**:

# ftpquota --show-records --type=limit --table-path=/etc/proftpd/ftpquota.limittab

ftpquota: (empty table)

**Para saber más**: ayuda del comando **ftpquota**, mediante **ftpquota –help**.

**No olvides recargar la configuración del servidor ProFTPd**: /etc/init.d/proftpd restart