

C.F.G.S. DESARROLLO DE APLICACIONES MULTIPLATAFORMA

MÓDULO:

Sistemas Informáticos

Unidad 9

Protocolo FTP

Redes

Uso del Protocolo FTP en modo
texto o desde una consola.

INDICE DE CONTENIDOS

1.	Tipos de conexión	3
2.	Conexión.	3
3.	Transmisión de datos.	4

1. Tipos de conexión

Control y Datos

Para iniciar una sesión FTP, aunque no nos demos cuenta, nuestro programa cliente usa dos conexiones simultáneas. Una de ellas sirve para el diálogo entre el cliente y el servidor, con el fin de que ambos se entiendan frente a las operaciones que deben realizar. Este tipo de conexión se denomina "**de control**". Por otro lado, tenemos la conexión de intercambio de archivos propiamente dicha, a la que denominamos conexión "**de datos**".

Puertos

Por lo general, y de manera estándar a nivel mundial, el protocolo FTP usa el **puerto 21**. Sin embargo, este puerto es el que **se utiliza para la transmisión de las señales de control** y no para la **transmisión de datos que usa generalmente el puerto número 20**.

Como dato significativo, es preciso señalar que el protocolo FTP permite comunicaciones "full duplex" es decir, que se pueden enviar y recibir datos al mismo tiempo.

Formatos

Existen varios formatos de transferencia de archivos, aunque hablaremos únicamente de los conocidos como ASCII o Binario.

Por lo general, siempre que se transfieran archivos de texto, aquellos que pueden abrirse por ejemplo con un Bloc de notas o un procesador de textos, deberán transmitirse en modo ASCII.

Si por el contrario, subimos imágenes, programas ejecutables, archivos comprimidos,...deberemos usar el formato Binario. Muchos clientes de FTP permiten predefinir qué archivos deberán tratarse como ASCII y suelen tener ya incluidos los más usuales como TXT, HTML, PHP y otros. El resto serán tratados en modo Binario.

2. Conexión.

Antiguamente o quizá no tanto si tenemos en cuenta la gran utilidad que supone el manejar un sistema Linux desde la línea de comandos, el uso del protocolo FTP en modo texto o desde una consola era y es algo muy habitual. Si en alguna ocasión se nos presenta la situación de tener que conectar con un servidor FTP y no tenemos a mano un cliente en modo gráfico (más sencillo), tampoco será demasiado complicado poder conectar, enviar y recibir datos.

Pongamos unos sencillos ejemplos.

Para conectar rápidamente, tan solo tenemos que escribir lo siguiente:

ftp dirección_IP del servidor

dicho de una manera real:

ftp 192.168.1.10

Donde el equipo 192.168.1.10 es el que contiene el servidor FTP.

Otra manera de hacerlo podría ser escribiendo únicamente "ftp" para dejar la consola en modo FTP (es decir, el cliente activado) y así poder escribir los comandos que necesitemos. Evidentemente, antes de poder escribir comandos, debemos conectar. Para ello, debemos escribir el comando OPEN, que nos preguntará por la dirección del servidor al que queremos acceder o pedir acceso:

```
[fer@servidor fer]$ ftp
ftp> open
<to> 192.168.1.10
```

Vemos que el cliente nos pregunta "hacia donde" (to) y le escribimos la dirección del servidor al que queremos conectar, en este caso 192.168.1.10. Una vez pulsemos ENTER, lo lógico es que se nos pida un nombre de usuario y contraseña. Le facilitamos esos datos y ya estamos conectados:

```
Name (192.168.1.10:fer): fer
331 Password required for fer.
Password:
230 User fer logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

3. Transmisión de datos.

Anteriormente detallamos la manera en que podemos acceder a un servidor FTP en modo texto. Ahora veremos muy específicamente algunos comandos:

cd: cambiar de directorio.

ls o dir: visualizar contenido de directorios.

get: "coger" archivos.

put o send: subir archivos.

delete: borrar archivos.

bye: desconectar

Comando CD.

Si al conectar al servidor no estamos en el directorio que queremos estar, usaremos el comando CD para poder navegar por el árbol de directorios.

Si quisiéramos entrar en un directorio llamado "programas" deberíamos escribir:

```
cd programas
```

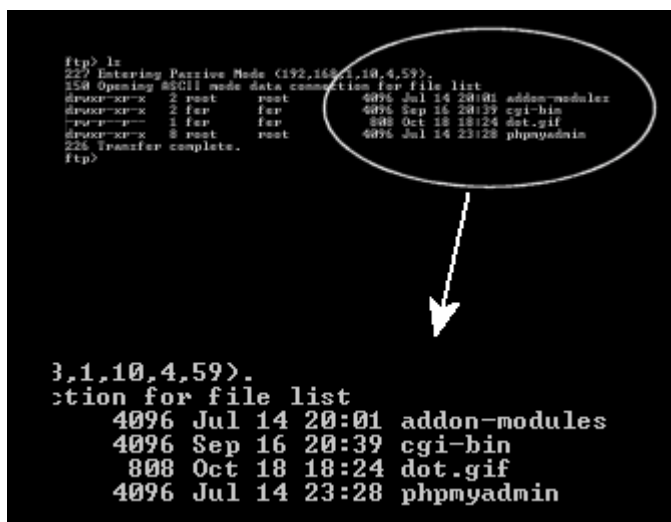
Si quisiéramos ahora descender al directorio que estábamos, deberíamos escribir:

```
cd .. (o también) cdup
```

Estas pautas siempre son las mismas, de modo que entrar y salir de directorios ya no es un misterio.

Comando LS o DIR.

Una vez conectados al servidor, lo primero que haremos será listar su contenido para empezar a "movernos". Como ya hemos dicho, podemos usar el comando LS (Unix) o DIR (DOS/Windows). Veamos un ejemplo:



```
ftp> ls
220 Entering Passive Mode (192.168.1.10,4,59).
150 Opening ASCII mode data connection for file list
drwxr-xr-x  2 root  root   4096 Jul 14 20:01 addon-modules
drwxr-xr-x  2 fer   fer    4096 Sep 16 20:39 cgi-bin
-rw-r--r--  1 fer   fer    808 Oct 18 18:24 dot.gif
drwxr-xr-x  8 root  root   4096 Jul 14 23:28 phpmyadmin
225 Transfer complete.
ftp>
```

3.1.10.4.59>. tion for file list

4096	Jul 14 20:01	addon-modules
4096	Sep 16 20:39	cgi-bin
808	Oct 18 18:24	dot.gif
4096	Jul 14 23:28	phpmyadmin

En el gráfico anterior, vemos un ejemplo de lo que aparece al escribir DIR o LS. En este caso se nos presentan 3 directorios y un archivo. El archivo es una imagen (dot.gif) y que intentaremos recoger a nuestro disco duro local, borrar, volverla a subir al servidor.

Comando GET.

Para poder "traernos" ese archivo a nuestro PC, usaremos el comando GET. Debemos tener en cuenta que el directorio de nuestro disco local donde se guardará el archivo, será el directorio donde estemos situados en ese momento. Así, si nuestra sesión FTP se efectúa desde una ventana de Windows, y en la ventana aparece por ejemplo C:\TEMP\ARCHIVOS, el fichero DOT.GIF, se guardará en el directorio TEMP\ARCHIVOS. Veamos qué es lo que sucede.

```
ftp> get dot.gif
200 PORT command successful
150 Opening ASCII mode data connection for dot.gif
226 Transfer complete.
ftp: 808 bytes recibidos en 0,00Segundos 808000,00
```

Vemos que es fácil de interpretar; nos hemos descargado el archivo DOT.GIF que ocupa 808 bytes.

Comando DELETE.

Si no queremos mantener el archivo DOT.GIF en el servidor FTP, podemos borrarlo, aunque recuerda que solamente podemos borrar archivos, si el administrador nos ha dado permiso para ello:

```
ftp> delete dot.gif
250 DELE command successful.
ftp> _
```

Ya está. Si ahora hiciéramos uso del comando DIR o LS, podríamos verificar que el archivo ya no está en el servidor.

Comando PUT o SEND.

Por último, nos queda por repasar el comando PUT o SEND. Ambos realizan la misma operación: poner un archivo de nuestro disco duro en el servidor. Veamos qué sucede:

```
ftp> send dot.gif
200 PORT command successful
150 Opening ASCII mode data connection for dot.gif
226 Transfer complete.
ftp: 808 bytes enviados en 0,00Segundos 808000,00KB
ftp> ls
200 PORT command successful
150 Opening ASCII mode data connection for file list
addon-modules
phpmyadmin
cgi-bin
dot.gif
226 Transfer complete.
ftp: 45 bytes recibidos en 0,00Segundos 45000,00KB
```

Lo primero que hemos hecho ha sido "send dot.gif" es decir, poner el archivo dot.gif en el servidor. El servidor nos responde que la transferencia se ha realizado con éxito (226 Transfer complete).

Seguidamente hacemos un LS (recuerda que es lo mismo que un DIR en servidores Unix) y se muestran los archivos y directorios que hay en servidor dentro de esa ubicación. Comprobamos que efectivamente, dot.gif se encuentra entre la lista.