

UT3: SERVICIO FTP: PROTOCOLO DE TRANSFERENCIA DE ARCHIVOS.

1. Introducción al servicio FTP
2. Funcionamiento del servicio FTP.
 - 2.1. Modos de conexión: Modo activo y modo pasivo
 - 2.2. Tipos de transferencia: binaria/ASCII
3. Formas de acceso al servidor FTP. Tipos de usuarios.
4. Conexiones y transferencias seguras: Protocolo SFTP.
5. Parámetros básicos de la configuración de un servidor FTP
6. Parámetros básicos de la configuración de los clientes FTP
7. Programas clientes FTP
 - 7.1. Comandos **ftp**
 - 7.2. Comando **sftp**
 - 7.2.1. 7.3.. Herramientas gráfica **gftp**
 - 7.3. Uso del navegador como cliente ftp
 - 7.4. Otros: PUTTY. Comando **wget**
8. Parte_I: Configuración del servidor **vsftpd** en Linux-CentOS 6.
9. Parte_II: Configuración del servicio FTP de **IIS** (Internet Information Service).

1. Introducción al servicio FTP

El servicio FTP o protocolo de transferencia de archivos pertenece a la capa de aplicación del modelo OSI. Proporciona un mecanismo estándar de transferencia de archivos entre sistemas, a través de TCP/IP:

Los primeros servidores FTP datan del año **1971**, siendo anteriores al Web en más de 10 años. Este servicio se creó por la necesidad de publicar documentos y software para que se pudieran obtener de forma fácil, desde otros ordenadores.

En el servidor FTP los ficheros están organizados en una estructura de directorios; los usuarios se autentican en el servidor y se conectan desde una red (Internet o Intranet) a través de programas clientes FTP, pueden desplazarse por la estructura de directorios a la que estén autorizados para localizar los archivos que interesen y descargarlos desde el servidor, a sus máquinas locales. Si les está permitido, podrán subir archivos al servidor.

La configuración del servicio requiere establecer el directorio en el servidor que ejercerá como raíz del **sitio FTP**. El administrador del sitio organizará la información del sitio: estructura de directorios, carpetas y archivos con los permisos adecuados, para el tipo de usuario que accede.

Los servidores FTP se usan principalmente para:

- subir las páginas web a un servidor o hosting.
- El acceso libre a archivos, denominado *FTP anónimo*. Los FTP anónimos son servidores FTP a los que puede conectarse cualquiera empleando como usuario *anonymous* y como password cualquier dirección de correo (servirá cualquier secuencia de caracteres que incluya "@").
- Hacer copias de seguridad (backup).

No obstante, el servicio FTP presenta una serie de deficiencias en cuanto a seguridad, ya que:

- Utiliza el mecanismo "normal" de autenticación de usuarios a través de un nombre (login) y una contraseña (password), con lo que el servidor no puede garantizar que el usuario es quien dice ser.
- Transfiere las contraseñas en texto plano (si cifrar) por lo que cualquier herramienta de captura de tramas por la red (sniffer), podría obtenerlas.
- No cifra la propia sesión en sí misma, por lo que las transferencias de archivos son, también, en texto plano.

2. Funcionamiento del servicio FTP.

El servicio FTP está basado en la arquitectura **cliente-servidor** por lo que necesita:

- ✓ un **servidor FTP** que se esté ejecutando en la máquina remota, y que por defecto, estará escuchando en los **puertos 20 y 21 TCP**:
 - el puerto **20** se utiliza para la **transmisión de datos**
 - el puerto **21** para la **transferencia de órdenes** (comandos de control).
- ✓ un **cliente FTP** para conectarnos al servidor.

2.1. Modos de conexión: modo activo y modo pasivo

Durante una conexión FTP se abren 2 canales de transmisión:

1. Un canal de comandos (canal de control).
2. Un canal de datos (para la transferencia de archivos)

La conexión FTP desde el cliente al servidor se puede realizar de 2 modos:

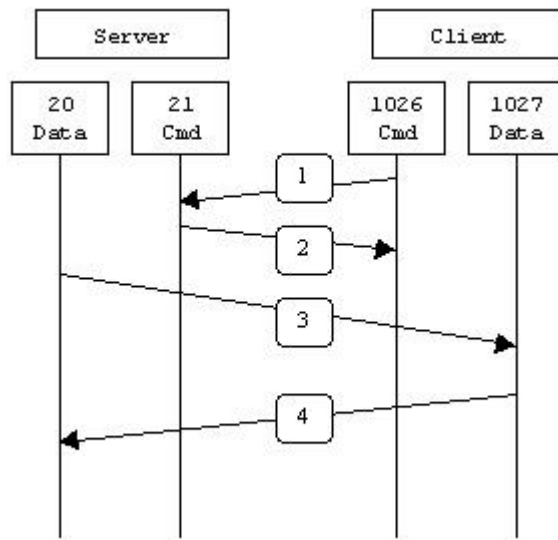
- **Modo activo:** el cliente no negocia con el servidor cuál es el puerto utilizado para la transmisión de datos. La transmisión se hace desde el puerto 20 del servidor al puerto indicado por el cliente.
- **Modo pasivo:** el cliente negocia con el servidor cuál es el puerto utilizado para la transmisión de datos. La transmisión se hace desde un puerto no privilegiado (>1024) elegido por el servidor, al puerto indicado por el cliente.

La conexión en modo activo es la forma nativa de FTP, de establecer la conexión. En este caso:

- Se establece la conexión desde el cliente, desde un nº de puerto no privilegiado ($N > 1024$), con el puerto 21 del servidor
- En esta conexión, el cliente comunica al servidor, qué puerto va a utilizar para recibir/transferir los datos (normalmente $N+1$).
- El servidor desde el puerto 20 se conecta en el puerto indicado en el cliente, ($N+1$), para realizar la transferencia solicitada por el cliente

Aquí tenemos un ejemplo de conexión:

1. El puerto de control del cliente (1026) se conecta al puerto de control del servidor (21) y envía el número del puerto de datos (1027)
2. El servidor responde con un ACK al puerto de control del cliente
3. El servidor inicia una conexión entre su puerto de datos (20) y el puerto de datos del cliente (1027)
4. El cliente responde con un ACK al servidor

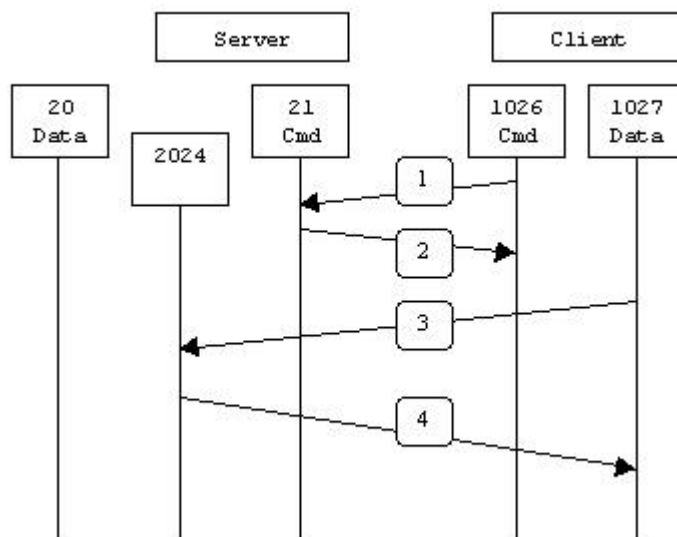


El modo activo tiene un inconveniente, ya que si el cliente está protegido por un cortafuegos, el servidor no podrá establecer una conexión al puerto de datos aleatorio del cliente (el que está por encima de 1024). Para solucionar este detalle hay otra forma de conexión llamada: FTP pasiva.

La conexión en modo pasivo:

- El cliente abre dos puertos aleatorios no privilegiados ($N > 1024$ puerto de control y $N+1$, puerto de datos)
- El cliente inicia siempre la conexión desde el puerto $N > 1024$ y se dirige al puerto 21 del servidor, para solicitarle un puerto superior a 1024 para la transferencia de datos (comando PASV)
- Luego, el servidor abre un puerto aleatorio no privilegiado ($P > 1024$) y se lo envía al cliente (comando PORT)
- Cuando el cliente recibe la respuesta ($P > 1024$), establece la conexión con dicho puerto P , desde $N+1$, para la transferencia de datos.

En este modo, el problema del cortafuegos se traslada al servidor, pero se soluciona limitando el nº de puertos accesibles para este modo de conexión.



En el ejemplo:

1. El puerto de control del cliente (1026) se conecta al servidor al puerto de control (21) y envía el comando PASV
2. El servidor responde al cliente con el puerto 2024, el puerto aleatorio de datos abierto para conexiones de datos
3. El cliente inicia una conexión desde su puerto de datos (1027) al puerto de datos del servidor (2024)
4. Finalmente, el servidor envía de vuelta un ACK al puerto de datos del cliente

2.2. Tipos de transferencia: binaria/ASCII

El protocolo FTP soporta los tipos de archivos: ASCII y binarios.

- El **código ASCII** se utiliza para representar caracteres y texto plano.
- El **código binario** se utiliza para representar todo lo que no sea texto (imágenes, audio, video, archivos ejecutables,...).

El modo de transferencia ASCII es el utilizado por defecto en la transferencia de datos, se utiliza cuando queremos enviar archivos de texto, HTML, ... si queremos transferir archivos binarios (programas, archivos comprimidos, multimedia,...) debemos usar el modo binario. Los programas específicos de transferencia de archivos detectan el tipo de archivo a transferir y no es necesario indicar el tipo.

3. Formas de acceso al servidor FTP. Tipos de usuarios.

La gestión de usuarios dentro de un servidor FTP es una de las tareas más importantes, ya que de ella depende la seguridad del contenido del servidor. Las partes del sistema de archivos a las que el usuario podrá acceder, y las operaciones que podrá realizar dependerá de los privilegios que se le hayan otorgado, en el servidor.

El acceso al sistema de archivos del servidor está limitado según el tipo de usuario que se conecta. El acceso puede ser: **privado** o **anónimo**, es decir, a un servidor FTP se puede acceder con 2 tipos de usuarios:

- **Usuarios autenticados:** son aquellos que tienen **cuenta en el servidor**, (previamente se habrá creado). Se conectan utilizando el nombre y password de dicha cuenta. Son conocidos como usuarios locales del servidor. Acceden a la parte del sistema de archivos que comienza en su directorio de conexión, con los permisos que tengan asignados sobre el sistema de archivos.
- **Usuarios anónimos:** no tiene una cuenta particular en el servidor, utilizan una cuenta genérica que suele llamarse **cuenta ftp**, y por tanto, no tiene los mismos privilegios que un usuario local. Acceden a un directorio público que se configura en el servidor (por ejemplo: el directorio **incoming** suele estar configurado para la subida libre de documentos o herramientas en servidores FTP anónimos). Se conectan con el nombre de usuario: **anonymous**, sin contraseña (o con una contraseña que represente una dirección de correo a@b).
- **Usuarios FTP:** utilizan cuentas en el servidor pero con sólo acceso exclusivo al servicio FTP. Se autentican en el servidor con cuentas que no les permite el inicio de sesión y por tanto no pueden trabajar en el servidor como pueden hacerlo los usuarios locales, es la forma habitual de conectarse a los servidores FTP.

4. Conexiones y transferencias seguras: Protocolo SFTP.

El servicio FTP no es seguro, ya que tanto la autenticación como la transferencia de información se realizan sin encriptar, por tanto es recomendable utilizar servicios más seguros:

- **SSH**, el cual implementa el protocolo **SFTP (Secure Shell File Transfer Protocol** o **Servicio Seguro de Transferencia de Ficheros)** proporciona un canal seguro en la transferencia de ficheros entre el cliente y el servidor. El servicio **SSH** (puerto **22**) debe estar bien configurado en el servidor FTP y hay que utilizar clientes seguros como: **sftp** para realizar la conexión segura con el servidor.
- **SSL**, (**Secure Socket Layer** o **Capa de Conexión Segura**) o **TLS (Transport Layer Security** o **Seguridad de la Capa de Transporte)**. El servicio FTP se apoya en estos protocolos para asegurar la conexión y transferencia de datos. Se utilizan mecanismos de cifrado, de clave pública y la utilización de certificados en el servidor. El puerto utilizado suele ser el **990** en lugar del habitual **21**.

5. Parámetros básicos de la configuración de un servidor FTP

Los parámetros básicos de configuración del servicio FTP determinan:

- La forma de acceso al sitio FTP por los usuarios: público o privado
- Los privilegios que tendrán sobre el sitio FTP: si los usuarios podrán acceder sólo a su directorio particular, o tendrán acceso a otros directorios
- Limitaciones respecto a la tasa de transferencia (tamaño máximo de transferencia de archivos)
- Nº de conexiones máximas permitidas y tiempo de conexión
- Gestión del espacio en disco (en el servidor)
- Si las conexiones serán seguras

6. Parámetros básicos de la configuración de los clientes FTP

La configuración en el cliente FTP no requiere parámetros especiales.

- Habrá que asegurar la correcta configuración de la red TCP/IP para que se pueda acceder desde ella a los distintos servidores de Internet y/o de una Intranet.
- Utilización de un programa cliente FTP para acceder al servidor FTP, bastará con indicar cuál es el servidor al que se quiere acceder mediante su dirección IP o su nombre (si está configurado un servidor DNS que lo pueda resolver) y realizar la autenticación en el servidor mediante acceso privado (login+password) o anónimo (mediante la cuenta de usuario anonymous). Existen programas cliente FTP en modo comando (comando **ftp** y comando **sftp**) o gráficos (**gftp**, **FileZilla**, **CuteFtp**,...). También se puede acceder desde el navegador utilizando la URL: <ftp://nombre-o-IP-del-servidor/> la parte **ftp://** indica que la conexión con el servidor se realiza a través del protocolo FTP:

7. Programas clientes FTP: comandos ftp, sftp, wget, navegadores y herramientas gráficas

Un programa cliente FTP es una aplicación que permite al usuario, desde su máquina local, conectarse a un servidor FTP remoto para solicitar o transferir archivos. Existen muchos programas o utilidades FTP, en modo texto y gráfico.

7.1. Modo texto: comando ftp y comando sftp (transferencia segura).

Permite registrarse en el servidor FTP y actuar desde un intérprete de comandos (**ftp>**).

```
$ftp nombre-servidor (ó IP)
Conected to nombre-servidor
Name: xxxxxxxxx
Password:*****
Remote system type is Unix
Using binary mode to transfer files
ftp>
```

También se puede realizar la conexión así:

```
ftp
ftp> open nombre-servidor (ó IP)
```

Una vez establecida la conexión con el servidor FTP, los comandos que se pueden utilizar desde este prompt **ftp>** permiten moverse y realizar operaciones tanto con los archivos del sistema local, como con los del remoto (servidor). Los comandos son tipo Unix, conocidos: **ls**, **pwd**, **cd**, **chmod** ...

Algunos de los **principales comandos FTP** son:

? (o help).	visualiza la lista o ayuda de los posibles comandos ftp.
pwd	muestra el nombre del directorio actual en el sistema remoto (servidor).
lpwd	lo mismo pero para el sistema local.
ls	Lista los contenidos del directorio remoto actual. Válidas todas la opciones vistas con el comando en UNIX.
dir	como ls.
ldir	como ls pero para el directorio local.
cd	cambia de directorio en el servidor. <code>cdup</code> y <code>cd</code> , es lo mismo.
lcd	cambia el directorio local actual.
mkdir	crea un directorio en el sistema remoto.
rmdir	elimina un directorio remoto.
rename	cambia el nombre de un fichero o un directorio en el sistema remoto.
delete	borra un fichero remoto.
mdelete	borra varios ficheros.
get	descarga un archivo.
mget	descarga una lista de archivos del servidor
put	escribe un archivo en el servidor (sube un archivo)
mput	escribe una lista de archivos en el servidor (no directorios).
open	abre una conexión FTP con un servidor FTP
close	termina la conexión actual pero no el cliente FTP.
quit o by o bye	termina la sesión y cierra la conexión

Para leer y grabar archivos es aconsejable tener como directorios activos tanto en la máquina local como en el servidor, los directorios desde donde vamos a leer y/o dejar los archivos:

get	toma un fichero desde el directorio actual en el sistema de archivos remoto, del servidor y lo copia al directorio actual en el sistema de archivos local (descarga archivos del del servidor) si en lugar de "archivo" ponemos una ruta tendremos que tener la misma ruta creada, en el ordenador local, para poder copiarlo. get archivo
mget	descarga todos los ficheros (no directorios) que se especifiquen, se pueden indicar comodines en los nombres de los archivos mget archivo1 archivo2 ...
put	copia un fichero del directorio local actual y lo deja en el directorio remoto actual (sube archivos al servidor) put archivo
mput	sube todos los ficheros que se especifiquen (no directorios), se pueden indicar comodines en los nombres de los archivos mput archivo1 archivo2 ...

Comandos para salir al intérprete: !

! este comando permite salir temporalmente al intérprete de comandos para realizar comandos **en la máquina local**, para volver a la sesión ftp, escribiremos **exit**. Sin salir al intérprete también podemos realizar comandos sobre la máquina local con **!comando**

Por ejemplo:

```
ftp> !ls -l visualiza el contenido del directorio activo en la máquina local
ftp> !pwd visualiza cuál es el directorio activo en la máquina local .
```

7.2. Comando **sftp**.transferencia segura.

La utilidad **sftp** puede ser usada para abrir una sesión segura interactiva de FTP. Es similar a ftp excepto que ésta utiliza una conexión cifrada segura.

Requiere de la instalación y configuración del servicio **SSH**.

ssh es un programa que permite acceder a otro ordenador a través de la red, provee autenticación y comunicaciones seguras sobre canales inseguros. Utiliza el puerto **22** que tendrá que abrirse en el firewall del servidor FTP.

La sintaxis general es:

sftp [nombre-usuario@hostname](#)

Donde hostname será el nombre del servidor FTP (tendrá que poder resolverse vía DNS) o su dirección IP, nombre-usuario será el nombre de la cuenta con la que se quiere conectar en el servidor. Si no se indica nombre de usuario, **sftp** entenderá que se quiere conectar con la cuenta de usuario con la que se está en el equipo local. Se puede conectar utilizando la cuenta del usuario **root** ya que la comunicación en este caso va cifrada.

Una vez autenticado el usuario en el servidor, podrá utilizar un conjunto de comandos similar al conjunto utilizado por el comando FTP.

Un ejemplo de uso:

```
[usuario1@localhost usuario1]$ sftp servidor.dominio.es
```

```
Connecting to servidor.dominio.es...
```

```
usuario1@servidor's password: sftp> help
```

Comandos disponibles:

help	
cd path	Change remote directory to
'path' lcd path	Change local directory to 'path'
chgrp grp path	Change group of file 'path' to 'grp'
chmod mode path	Change permissions of file 'path' to
'mode' chown own path	Change owner of file 'path' to 'own'
help	Display this help text
get remote-path [local-path]	Download file
lls [ls-options [path]]	Display local directory
ln oldpath newpath	Symlink remote file
mkdir path	Create local directory
lpwd	Print local working directory
ls [path]	Display remote directory
lls [path]	Display remote directory
listing lumask umask	Set local umask to 'umask'
mkdir path	Create remote directory
put local-path [remote-path]	Upload file
pwd	Display remote working
directory exit	Quit sftp

quit	Quit sftp
rename oldpath newpath	Rename remote file
rmdir path	Remove remote
directory rm path	Delete remote file
symlink oldpath newpath	Symlink remote file
version	Show SFTP
!command	Execute 'command' in local shell
!	Escape to local shell
?	Synonym for help

Otros comandos básicos del servicio SSH:

Comando ssh:

Para iniciar una sesión en otra máquina usando **ssh**:

ssh user@hostname [command]

Comando scp:

El comando **scp** permite copiar ficheros entre dos máquinas, a través de la red. Utiliza **ssh** para la transmisión de la información, por lo que ofrece la misma seguridad que el **ssh**.

Un ejemplo de uso del scp para copiar desde la máquina local a una remota:

[usuario1 @localhost] **scp /tmp/file usuario1 @servidor.dominio.es:/tmp**

7.3. Herramientas gráficas cliente FTP.

Existen multitud de herramientas gráficas que facilitan tanto la labor de la administración del servidor FTP, como su acceso. Algunas de las más utilizadas son: **gftp, FileZilla, CuteFtp ...**

Estas herramientas abren una ventana gráfica desde donde podemos conectar a cualquier servidor FTP (de Internet o de nuestra Intranet), guardar la configuración de la conexión y desde donde nos podemos mover por el sitio, descargar o subir archivos.

La pantalla se divide en 2 zonas en las que se visualiza simultáneamente los contenidos de los directorios remotos en el servidor (normalmente la parte derecha de la ventana) y locales (izquierda), simplemente hay que hacer doble clic o arrastrar y soltar ficheros para dejarlos en uno u otro lado. También podemos añadir a **Marcadores** la dirección de los sitios FTP que nos parezcan interesantes: información de inicio de sesión, sobre las conversaciones que se han desarrollado entre nuestra máquina y cada sitio FTP.

7.4. Uso del navegador web como cliente FTP

El navegador puede utilizarse como cliente FTP. Basta con teclear la URL del servidor FTP correspondiente y acceder a los archivos para los que se tenga permisos :

ftp://ftp.server.com, donde ftp.server.com es el nombre (o IP) del servidor FTP. Podemos utilizar cualquier otro navegador Web.

Por ejemplo, para acceder al servidor ftp de rediris.es, escribiremos: **ftp://ftp.rediris.es**

7.5. Otros:

7.5.1 PuTTY

PuTTY (Puerto Unico de Terminal), es un emulador gratuito de terminal que soporta SSH y muchos otros protocolos, es muy útil cliente SSH para conectar con un servidor Unix o Linux a través de SSH. Ofrece una interfaz gráfica de configuración muy sencilla con múltiples opciones:

- Guardar las preferencias de conexión para establecerla rápidamente en el futuro
- Respuestas de puertos
- Soporte Ipv6
- Soporte SCP y SFTP

Además de ser para Windows, PuTTY ha sido portado a sistema Linux y Mac OS X. Es de código abierto.

7.5.2. Comando **wget** (Linux)

Si sabemos dónde se encuentra un archivo en la red, existe una forma más eficaz de descargarlo que abrir una sesión FTP, y es utilizando el comando **wget** y la dirección del sitio FTP: permite la conexión como usuario anónimo o como usuario si le proporcionamos nombre y contraseña en la línea de comandos:

```
wget ftp://ftp.redhat.com/pub/redhat/news/checkgroups
Connecting to ftp.redhat.com
Login in as anonymous
100%[=====]
```

Como sabemos, **wget** también permite la descarga desde sitios web:
wget http://www.redhat.com/.....

8. Parte_I: Configuración del servidor vsftpd en Linux-CentOS 6.

9. Parte_II: Configuración del servidor FTP en Windows Server 2008: servicio FTP de IIS (Internet Information Server).