

## Unidad 3: Instalación y configuración del servicio FTP (Primera parte)

**Desarrollo de la unidad:** 8 horas

**Conceptos:** Funcionamiento del protocolo FTP, Clientes FTP: comando, gráficos, navegadores  
Servidores de FTP sobre Windows y sobre GNU/Linux, TFTP y SFTP

**Ejercicios y Prácticas:**

Conectarse a un servidor FTP con distintos clientes

Instalar y configurar clientes FTP

Instalar y configurar servidores FTP

Diagnóstico y detección de errores mediante Ethereal

### 1. INTRODUCCIÓN: EL PROTOCOLO FTP

El FTP (*File Transfer Protocol*), es el Protocolo de Transferencia de Ficheros más utilizado en la red, aunque la posibilidad de enviar y recibir ficheros mediante http o mediante otros servicios tipo p2p (kazaa, eMule) están reduciendo poco a poco su uso.

Cuando se establece una comunicación mediante FTP entre dos máquinas ha de superarse una fase previa de autenticación basada en un fichero de contraseñas. Sin embargo, esta comunicación **establecida no es segura, ya que información viaja sin codificar** ni las contraseñas ni los datos, tampoco tiene capacidades de filtrado. Aunque resulta muy difícil tomar el control de una máquina, sólo con la conexión vía FTP, un servidor mal **configurado puede ser fácilmente atacado alterando ficheros privados o de administración o mediante peticiones masivas que provoquen una denegación de servicio.**

Pese a todo, constituye una de las herramientas más útiles para el intercambio de ficheros entre diferentes ordenadores y es la **forma habitual de publicación (subir archivos) en Internet.**

Aunque puedan contemplarse otras posibilidades, hay dos tipos fundamentales de acceso a través de FTP:

- **Acceso anónimo**, cuando el contacto con la máquina lo realiza un usuario sin autenticar y sin ningún tipo de privilegio en el servidor. En ese caso, el usuario es confinado a un directorio público donde se le permite descargar los archivos allí ubicados pero sin posibilidad de escribir ningún fichero. No se le permite, normalmente, subir de nivel y listar los contenidos de los directorios de nivel superior.

- **Acceso autorizado**, cuando el usuario que solicita la conexión tiene una cuenta con ciertos privilegios en el servidor y, tras autenticarse, se le confina a su directorio predeterminado desde donde puede descargar ficheros y, si la política del sistema se lo permite, también escribir, aunque normalmente se limita su espacio mediante una cuota de disco. Puede estar autorizado a recorrer parte del árbol de directorios y listar su contenido o escribir en ellos, dependiendo del tipo de privilegios que posea.

La arquitectura del servicio se **basa en un modelo cliente-servidor**, donde los clientes serán usuarios de equipos de la red y el servidor se instalará en su sistema.

### 2. El servicio FTP

Para usar FTP para transferir archivos entre dos ordenadores, cada uno de estos ordenadores deben soportar sus respectivos papeles, es decir, uno debe ser el cliente FTP y el otro el servidor FTP. El cliente envía comandos al servidor y éste los lleva a cabo. Estos comandos pueden ser bajar o subir ficheros al servidor, cambiar de directorio o crear un directorio en el servidor, etc. FTP usa TCP como protocolo de transporte para toda la comunicación e intercambio de datos entre cliente y servidor. TCP es un protocolo orientado a conexión. Esto quiere decir que se debe establecer una conexión entre los dos nodos antes de que empiece el intercambio de datos entre ambos. Esta conexión estará activa durante toda la sesión FTP. Una transmisión orientada a conexión se caracteriza por incorporar

mecanismos de detección y corrección de errores y por su fiabilidad. Por eso se puede decir que FTP es muy fiable.

FTP usa dos conexiones TCP distintas, una conexión TCP de control (Puerto 21) y otra conexión TCP de transferencia de datos (Puerto 20). La conexión de control se encarga de iniciar la comunicación entre el cliente y el servidor y mantenerla activa, mientras que la conexión de transferencia de datos tiene como objetivo enviar datos entre cliente y servidor. La conexión de transferencia de datos existe sólo cuando hay datos que transmitir. Esta conexión se cierra cuando la transferencia de datos ha finalizado, mientras que la conexión de control sigue abierta durante toda la sesión.

El servicio FTP tiene dos modos de funcionamiento el modo estándar o **activo** que utilizan el puerto 20, para transferencia de datos y el modo **pasivo** que utiliza un puerto al azar (> 1024) y que ofrece mayor seguridad al no necesitar configuraciones adicionales en el cortafuegos del cliente y del servidor. En el primero es el cliente quien tiene que abrir el puerto para recibir la transferencia del servidor y en el segundo es el servidor que abre el puerto.

Cuando el servidor FTP está activo en una máquina, está constantemente escuchando en el puerto 21 esperando la llegada de una petición de conexión de un cliente (Tiene el puerto abierto). Cuando un cliente quiere iniciar una sesión FTP en una máquina remota se sigue el siguiente procedimiento para abrir una conexión de control:

1. Se asigna un número de puerto TCP mayor de 1023 para la aplicación cliente.
2. TCP crea un paquete que incluye la siguiente información:
  - Dirección IP del cliente.
  - Número de puerto del cliente, que será mayor de 1023.
  - Dirección IP del servidor.
  - Número de puerto del servidor, en este caso el 21.
3. El paquete es enviado a través de la red hasta la dirección IP del servidor.
4. El servidor recibe el paquete a través del puerto 21 y procesa los datos.
5. El servidor FTP envía un asentimiento al cliente usando la información que el cliente le envió como es la dirección IP y número de puerto del cliente. Para cada transferencia de archivos se establece una conexión independiente.

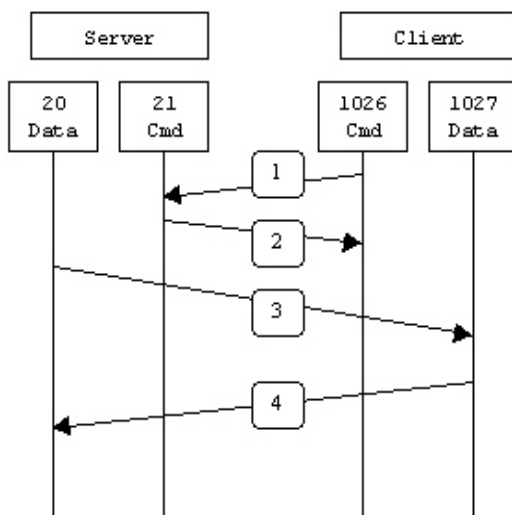


Fig. FTP modo activo

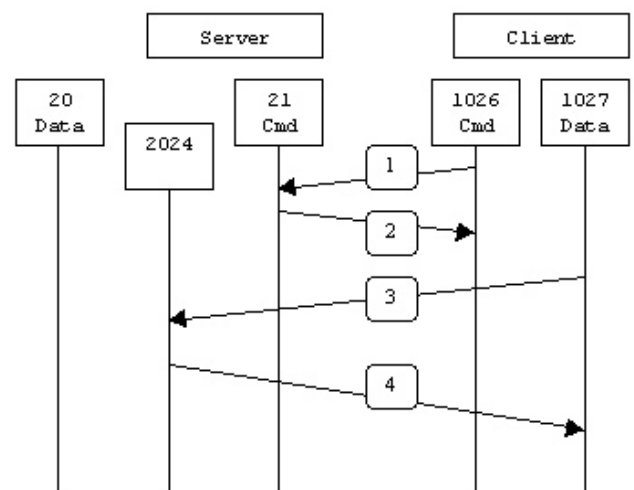
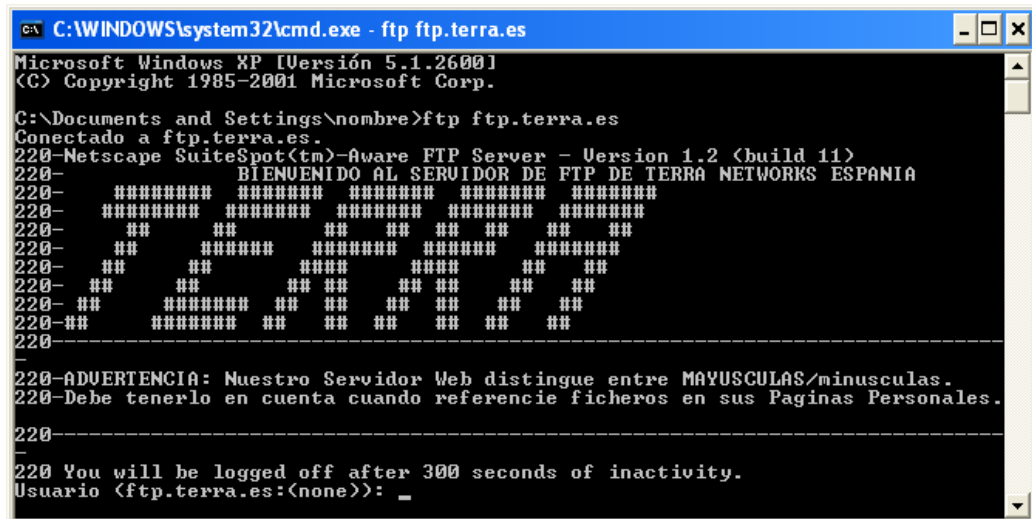


Fig. FTP modo pasivo

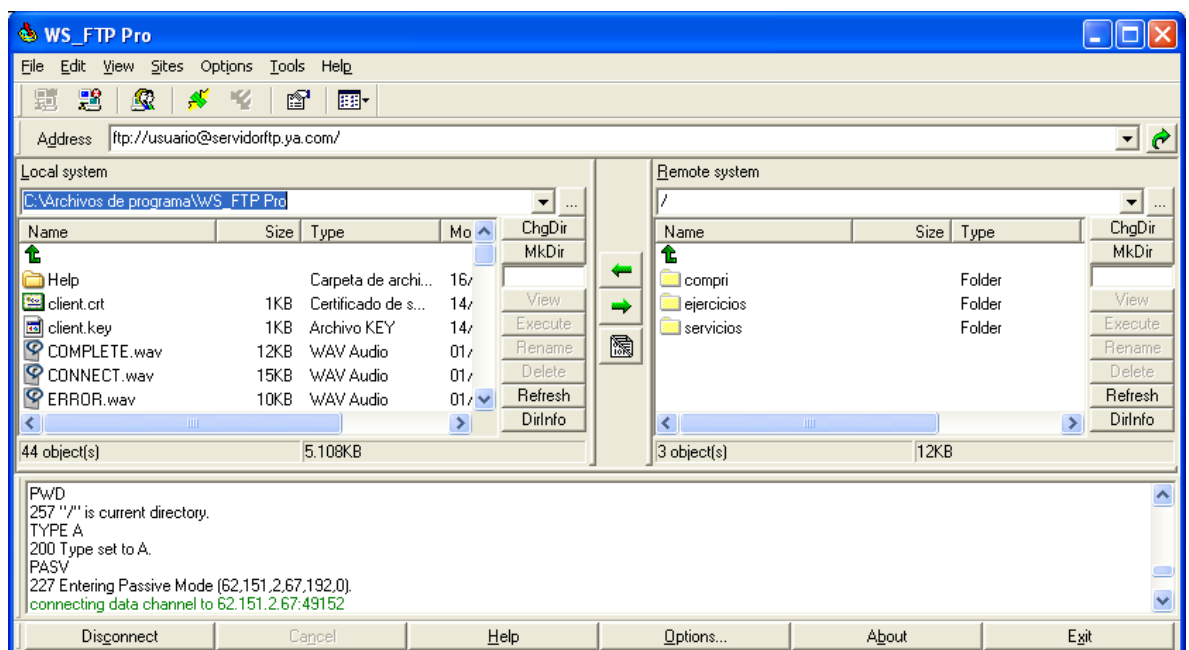
### 3. Clientes de FTP

## Existen tres tipos de clientes de FTP

- Clientes modo comando. Es el más estándar y está prácticamente incluidos en todos los sistemas operativos.



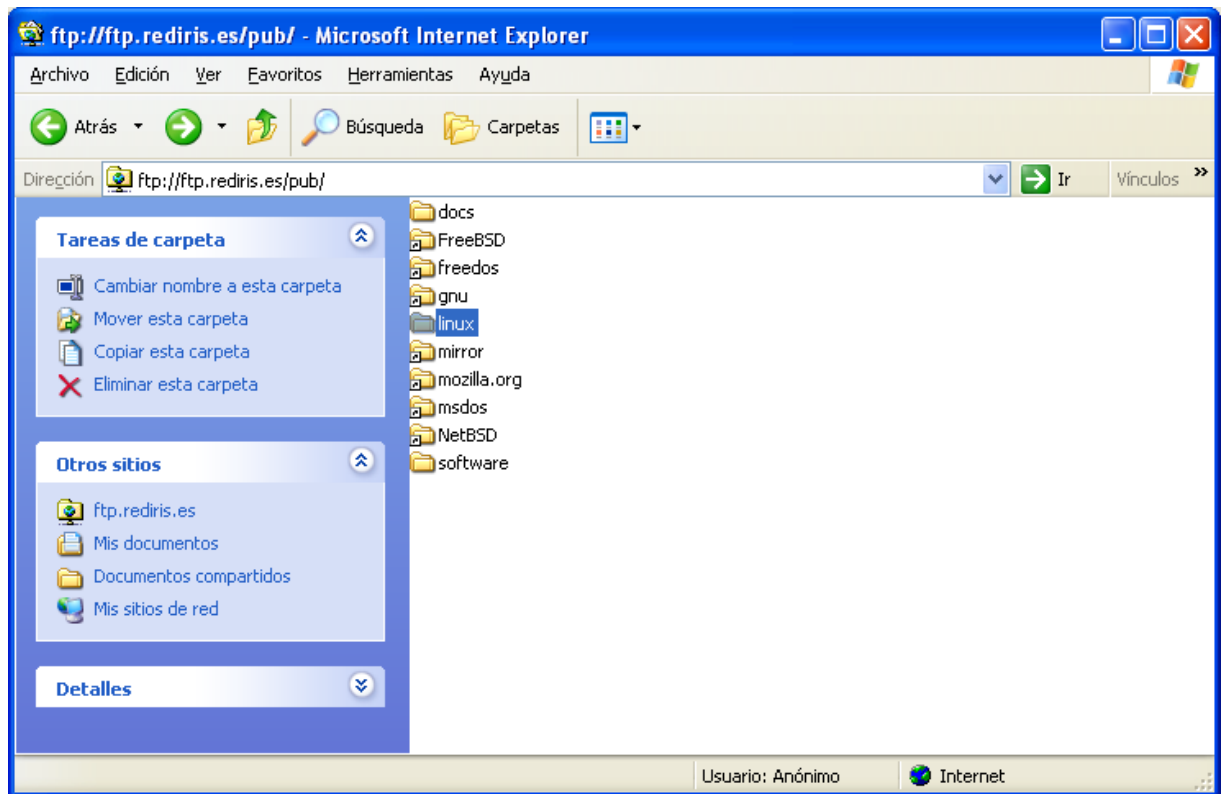
- **Clientes gráficos:** Internamente funcionan igual que el modo comando pero ofrecen un interfaz gráfico que facilita la transferencia de los ficheros sin necesidad de introducir las ordenes.



- **Navegadores:** La mayoría de los navegadores (Explorer, FireFox, Mozilla, Opera) incluyen un cliente ftp, normalmente bastante limitado, pero muy fácil de usar. Para invocarlo basta con incluir la dirección URL el protocolo FTP Ej.- <ftp://ftp.rediris.es>  
Si no vamos a utilizar una conexión anónima podemos incluir el nombre del usuario y la contraseña.

<http://usuario@servidor.terra.es> (Sólo usuario, el navegador nos solicitará posteriormente la contraseña)

<ftp://usuario:clave@servidor.terra.es> (Establece la conexión si el usuario y contraseña son correctos)



## 4. Modo Comando

El comando *ftp* constituye un cliente básico. Este recibe como argumento principal la identificación del servidor (nombre o número IP), además del número del puerto (por defecto 21). También se pueden indicar algunas opciones.

Si se logra la conexión con el servidor se realiza la autenticación. En cualquier caso, falle o no la autenticación o no se logre abrir la conexión, se activará el *prompt* del cliente de forma similar a:

```
ftp>
```

A partir de aquí se podrán ejecutar una gran cantidad de comandos del protocolo.

### LISTA DE COMANDOS MÁS HABITUALES EN FTP

• <b>open</b> [ <i>host</i> ] : abre una conexión FTP con el <i>host</i> especificado.
• <b>user</b> [ <i>login</i> ] : permite autenticarse como un usuario determinado.
• <b>ls</b>
• <b>dir</b> : muestran el contenido del directorio actual en el servidor en formato largo (similar a <code>ls -al</code> ).
• <b>get</b> [ <i>fichero</i> ] : traslada un fichero desde el servidor al cliente.
• <b>put</b> [ <i>fichero</i> ] : traslada un fichero desde el cliente al servidor.
• <b>mget</b> [ <i>patrón</i> ] : traslada todos los ficheros que satisfagan cierto patrón del servidor al cliente.
• <b>mput</b> [ <i>patrón</i> ] : traslada todos los ficheros que satisfagan cierto patrón del cliente al servidor.
• <b>binary</b> : establece el modo de transferencia binario (todo lo que no sea texto ASCII).
• <b>ascii</b> : establece el modo de transferencia para texto ASCII.
• <b>cd</b> [ <i>directorio</i> ] : cambia el directorio actual en el servidor.
• <b>prompt</b> : habilita o deshabilita el modo interactivo. Por ejemplo, cuando se transfieren a la vez más de un fichero utilizando un patrón, en el modo interactivo se pregunta para cada uno de ellos si se desea transferir o no.
• <b>lcd</b> [ <i>directorio</i> ] : cambia el directorio actual en el cliente.
• <b>!&lt;comando&gt;</b> : ejecuta un comando en un <i>shell</i> en el cliente.

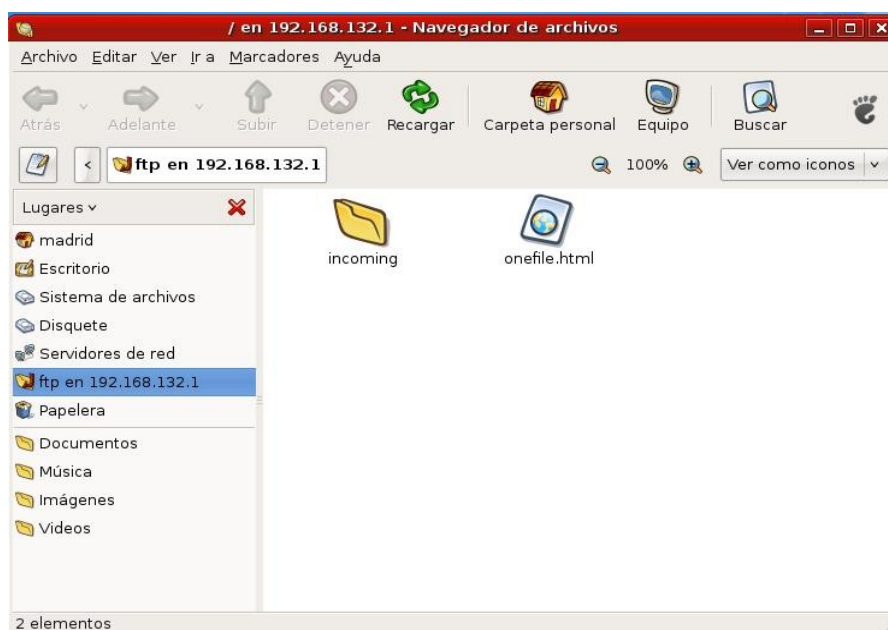
- **close** : cierra la conexión actual.
- **help** [comando] Ofrece información sobre un comando concreto o sobre la lista de comandos
- **exit** | **bye** | **quit** : Hace lo mismo cierran la conexión actual y terminan el programa.

*Si queremos enviar ficheros de texto planos, conviene activar el modo ascii para convertir automáticamente entre distintos formatos de texto (Ej- Windows y UNIX) en otro caso el modo binary es el más adecuado.*

### Modo sitio o ubicación de red.

Este es un modo menos conocido pero que facilita el trabajo del usuario y está incluido tanto en sistemas Windows como en GNU/Linux en escritorio GNOME

*Los exploradores de ficheros puede configurar una unidad o sitio de red para poder trabajar con un servidor FTP como si fuera una unidad o directorio más de nuestro sistema.*



## 5. SERVIDORES FTP.

Existen distintos programas que actúan como servidores de FTP, en entorno Windows tenemos este servicio incluido IIS ( Internet Information services) incluido en algunas versiones Windows Cliente (XP/Vista//) y Windows Server. En este entorno también podemos instalar el servidor FileZilla (<http://filezilla-project.org>) de software libre.

En entorno UNIX y GNU/Linux encontramos varias opciones siendo las más frecuentes: vsftpd (<http://vsftpd.beasts.org>) y proftpd (<http://www.proftpd.org>).

### 5.1 CONFIGURACIÓN DEL SERVIDOR VSFTPD

**Very Secure FTP Daemon** es un software utilizado para implementar servidores de archivos a través del protocolo FTP. Se distingue principalmente porque sus valores por defecto son muy seguros y por su sencillez en la configuración, comparado con otras alternativas como Wu-ftp. Actualmente se presume que VSFTPD es quizá el servidor FTP más seguro del mundo. Permite la utilización tanto de usuarios reales como usuarios virtuales extraídos de una base de datos.

#### **Ficheros de configuración.**

/etc/vsftpd.conf	Fichero de configuración principal.
/etc/vsftpd.ftputers	Lista de los usuarios sin permiso para conectarse al servicio FTP
/etc/vsftpd.user_list	Lista que definirá usuarios con permiso o sin él para conectarse al servicio FTP.
/etc/vsftpd.chroot_list	Lista de usuarios a encerrar o no dentro de su directorio de trabajo

#### **Procedimientos.**

Arranque el editor de texto de su preferencia y edite /etc/vsftpd/vsftpd.conf. A continuación analizaremos los parámetros a modificar o añadir según sea necesario. El significado y las opciones de todos los parámetros se puede consultar en el manual (\$man vsftpd.conf ).

##### **Parámetro anonymous\_enable.**

Este parámetro se utiliza para definir si se permitirán los accesos anónimos al servidor. Establezca el valor como valores YES o NO de acuerdo a lo que se requiera.

```
anonymous_enable=YES
```

##### **Parámetro local\_enable.**

Este parámetro es particularmente atractivo si se combina con la función de jaula. Establece si se van a permitir los accesos autenticados de los usuarios locales del sistema. Establezca el valor YES o NO de acuerdo a lo que se requiera.

```
local_enable=YES
```

##### **Parámetro write\_enable.**

Este parámetro establece si se permite el mandato "write" (escritura) en el servidor. Establezca el valor YES o NO de acuerdo a lo que se requiera.

```
write_enable=YES
```

**Parámetro ftpd\_banner.**

Este parámetro sirve para establecer el banderín de bienvenida que será mostrado cada vez que un usuario acceda al servidor. Puede establecerse cualquier frase breve que considere conveniente.

```
ftpd_banner=Bienvenido al servidor FTP de nuestra empresa.
```

**Estableciendo jaulas para los usuarios: parámetros chroot\_local\_user y chroot\_list\_file.**

De modo predefinido los usuarios del sistema que se autentiquen tendrán acceso a otros directorios del sistema fuera de su directorio personal. Si se desea recluir a los usuarios para que solo puedan moverse dentro de su propio directorio personal, puede hacerse fácilmente con el parámetro `chroot_local_user` que habilitará la función de `chroot()` y los parámetros `chroot_list_enable` y `chroot_list_file` para establecer el fichero con la lista de usuarios que quedarán excluidos de la función `chroot()`.

```
chroot_local_user=YES  
chroot_list_enable=YES  
chroot_list_file=/etc/vsftpd/vsftpd.chroot_list
```

Con lo anterior, cada vez que un usuario local se autentique en el servidor FTP, solo tendrá acceso a su propio directorio personal y lo que este contenga. No olvide crear `/etc/vsftpd/vsftpd.chroot_list`, ya que de otro modo no arrancará `vsftpd`.

**Control del ancho de banda.****Parámetro anon\_max\_rate.**

Este parámetro es utilizado para limitar la tasa de transferencia en bytes por segundo para los usuarios anónimos, algo sumamente útil en servidores FTP de acceso público. En el siguiente ejemplo se limita la tasa de transferencia a 5 Kb por segundo para los usuarios anónimos:

```
anon_max_rate=5120
```

**Parámetro local\_max\_rate.**

Este parámetro hace lo mismo que `anon_max_rate`, pero aplica para usuarios locales del servidor. En el siguiente ejemplo se limita la tasa de transferencia a 5 Kb por segundo para los usuarios locales:

```
local_max_rate=5120
```

**Parámetro max\_clients.**

Este parámetro establece el número máximo de clientes que podrán acceder simultáneamente hacia el servidor FTP. En el siguiente ejemplo se limitará el acceso a 5 clientes simultáneos.

```
max_clients=5
```

**Parámetro max\_per\_ip.**

*Este parámetro establece el número máximo de conexiones que se pueden realizar desde una misma dirección IP. Tome en cuenta que algunas redes acceden a través de un servidor proxy o puerta de enlace y debido a esto podrían quedar bloqueados innecesariamente algunos accesos. en el siguiente ejemp*



## OTROS PROTOCOLOS DE TRANSFERENCIA DE FICHEROS

### TFTP (Trivial File Transfer Protocol)

Es un protocolo sencillo (Trivial) para el envío de ficheros. Difiere de FTP estándar en dos formas:

- No hay necesidad de código ni contraseña de entrada.
- Utiliza el protocolo UDP.

Al utilizar UDP, TFTP no se garantiza la integridad de la comunicación y requiere de algoritmos adicionales para garantizar la integridad de los archivos. El mecanismo de control de errores se basa en un esquema stop and wait (para y espera) con confirmación enviando bloques de 512 bytes numerados desde 1.

TFTP utiliza el puerto 69 de UDP

TFTP se utiliza principalmente en estaciones o dispositivos sin disco para cargar sistema operativo, aplicaciones, archivos de configuración, etc. todo ello vía red, a partir de los datos almacenados en un equipo servidor. El protocolo es tan sencillo que se puede grabar en un chip de memoria EEPROM de la tarjeta Ethernet.

### SFTP (SSH File Transfer Protocol )

Es un módulo del servicio de conexión remota SSH y permite ordenes similares al FTP. Toda la información intercambia entre el cliente y el servidor, se produce a través de un subnivel de transporte SSL (Secure Socket Level) que encripta toda la información intercambiada, tanto los datos como las contraseñas. Normalmente admite las mismas ordenes que el protocolo FTP

Jerarquía de protocolos:

SFTP --> SSL --> TCP --> IP Protocolo de Nivel de Enlace

- El Servidor FTP en este caso, no es un servidor independiente, sino que constituye un subsistema del servidor principal del "SSH" (Secure Shell) Interprete de comandos seguro.
- Al ser un subsistema del SSH el servidor escucha en el puerto TCP 22.

Ejemplo:

```
$sftp [opciones] [usuario@]host[:fuente [destino] ]
```

Gran parte de los clientes de FTP permiten conexiones seguras (modo SFTP) con SSL por el puerto 22

### FTPS. ( FTP - secure )

FTPS (también referido como FTP/SSL) es la forma de designar diferentes formas a través de las cuales se pueden realizar transferencias seguras de ficheros a través de FTP utilizando SSL . Son mecanismos muy diferentes a los del protocolo SFTP (SSH File Transfer Protocol).

Existen dos diferentes métodos para realizar una conexión SSL/TLS a través de FTP. La primera y más antigua es a través del FTPS Implícito (Implicit FTPS), que consiste en cifrar la sesión completa a través de los puertos 990 (FTPS) y 998 (FTPS Data), sin permitir negociación con el cliente, el cual deberá conectarse directamente al servidor FTPS con el inicio de sesión SSL/TLS. El segundo método, que es el recomendado por el RFC 4217 y el utilizado por Vsftpd, es FTPS Explícito (Explicit FTPS o FTPES), donde el cliente realiza la conexión normal a través del puerto 21 y permitiendo negociar opcionalmente una conexión TLS.



### **BBFTP**

Se han diseñado distintos protocolos que ofrecen mayores rendimientos y posibilidades que el FTP estándar.

El BBFTP es un protocolo de transferencia optimizado para ficheros mayores de 2GB. Sus principales características son:

- Codifica user / passwd.
- Módulos de autenticación SSH y Certificado.
- Transferencia multi-stream (Abre varias conexiones para un envío).
- Compresión al vuelo (zlib). (Comprime y descomprime en marcha )
- Reintento automático.
- Simulación de Transferencia.
- Open-Source. (Código abierto, software libre)

#### **Inconvenientes:**

- No está muy difundido.
- Sólo se utiliza para el intercambio de datos masivo entre servidores.