

4.1. Introducción

Una de las principales ventajas que ofrecen las redes basadas en TCP/IP es la posibilidad de transferir información entre los equipos que forman parte de ellas. Existen múltiples servicios de red que permiten a los usuarios enviar ficheros de unos sistemas a otros, por ejemplo, servicios de correo electrónico (adjuntando archivos en los mensajes), servicios web (a través de hipervínculos que apuntan a ficheros localizados en un servidor web), servicios P2P (*Peer to Peer*) de descarga, servicios para compartir recursos en red (Samba/SMB/CIFS), etc. Algunos de los servicios nombrados anteriormente no están diseñados exclusivamente para transferir ficheros y no suelen estar optimizados para ello.

En este capítulo se explican servicios diseñados con el objetivo principal de transferir archivos entre diferentes sistemas, tratando en profundidad el basado en el protocolo FTP (*File Transfer Protocol*).

4.2. Servicio FTP

FTP es un protocolo de capa de aplicación diseñado para ofrecer un servicio estándar de transferencia de ficheros entre sistemas conectados a redes TCP/IP.

4.2.1. Características

FTP ofrece uno de los servicios más antiguos (en el año 2011 cumplió 40 años) de transferencia de ficheros y aun así se sigue utilizado en Internet y en redes corporativas.

Permite a los usuarios:

- Acceder a sistemas remotos y listar directorios y ficheros.
- Transferir ficheros desde o hacia el sistema remoto, es decir, subir (*upload*) o bajar (*download*) ficheros.
- Realizar acciones adicionales en el sistema remoto como renombrar, borrar, crear archivos y carpetas, cambiar permisos, descomprimir,....

Es un servicio fácil de mantener y configurar para los administradores, ofrece rapidez en la transferencia de ficheros y abstrae a los usuarios de los detalles de los sistemas operativos empleados. Existen múltiples implementaciones tanto libres como propietarias.

4.2.2. Componentes y funcionamiento

Su funcionamiento se basa en el modelo **cliente/servidor** y está formado por los siguientes componentes, Figura 4.1:

- **Clientes FTP.** Acceden al sistema de ficheros del equipo donde están instalados y establecen conexiones con los servidores FTP para subir o descargar archivos.
- **Servidores FTP.** Acceden al sistema de ficheros del equipo donde están instalados, manejan las conexiones de los clientes y en función de los privilegios definidos permiten la descarga y/o la subida de ficheros.

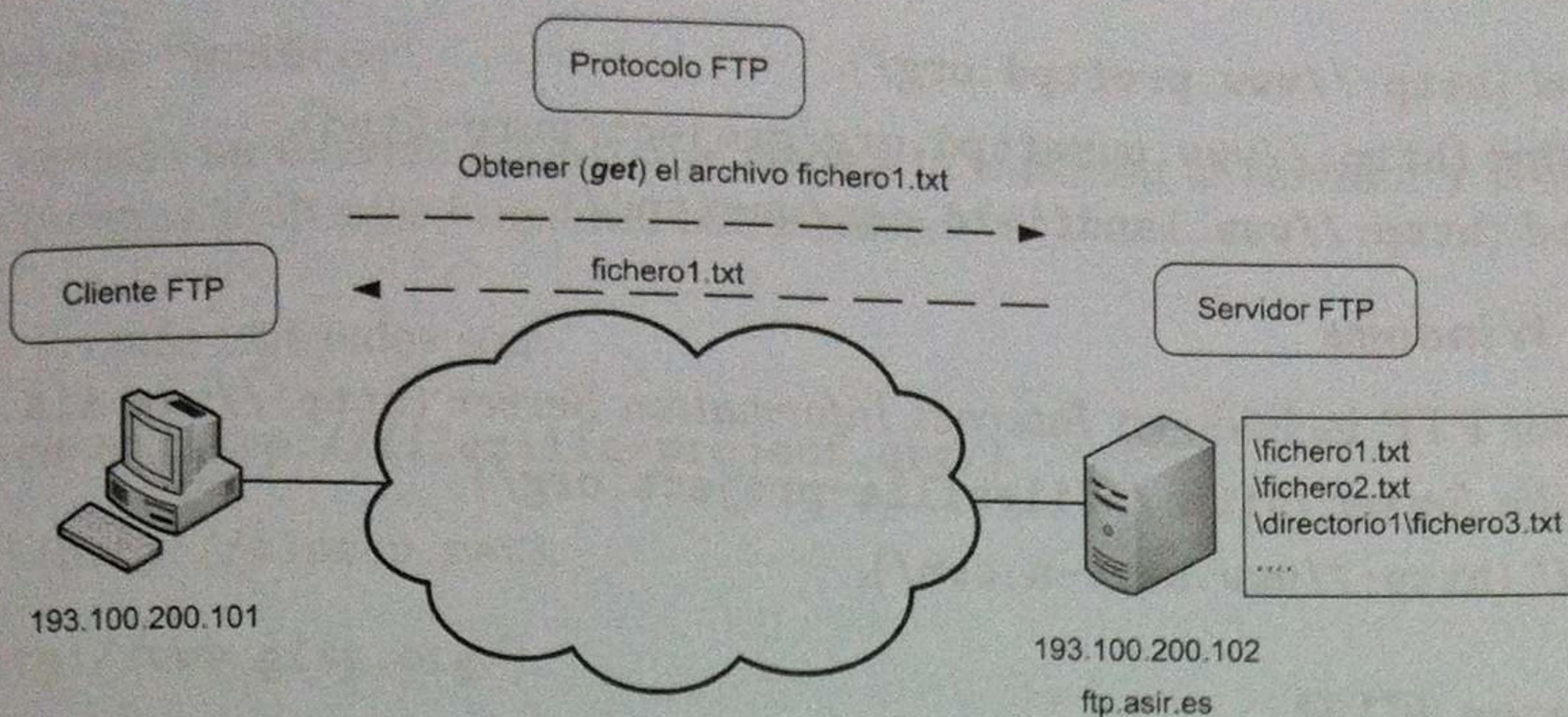


Figura 4.1: Componentes FTP

- **Protocolo FTP.** Conjunto de normas y reglas en base a las cuales “dialogan” los clientes y los servidores FTP. Usa TCP como protocolo de transporte.

◊ Actividad 4.1:

- Busca en Internet servidores FTP públicos y elige alguno (por ejemplo, ftp.rediris.es, ftp.debian.com,...)
- Accede a un equipo del aula, abre un terminal y utiliza el cliente FTP en línea de comandos para establecer una conexión como usuario anónimo al servidor FTP que has seleccionado. Por ejemplo, introduce `ftp ftp.rediris.es`. Como usuario emplea “**anonymous**” y deja la contraseña vacía. Una vez conectado al servidor ejecuta el comando `ls` para listar el contenido del directorio donde has accedido.
- Utiliza el navegador como cliente FTP para establecer una conexión como usuario anónimo al servidor FTP que has seleccionado. Inicia un navegador e introduce por ejemplo la URL [ftp://ftp.rediris.es](http://ftp.rediris.es). Observa que has accedido al mismo directorio que usando el cliente FTP en línea de comandos.

4.2.3. Servidores FTP

Un servidor FTP es un programa que atiende y procesa las conexiones de los clientes FTP y que puede acceder al sistema de ficheros del equipo donde está instalado permitiendo la subida y bajada de archivos. Ofrecen múltiples opciones de configuración para establecer privilegios de los usuarios, limitaciones de subida y descarga, tiempos de conexión y espera, etc.

Existen múltiples servidores FTP tanto para sistemas libres como para sistemas propietarios. Algunos de los más utilizados son:

- Sistemas *Linux/Unix*
 - *vsftpd* (<http://vsftpd.beasts.org/>).

- *proftpd* (<http://www.proftpd.org/>).
- *pure-ftpd* (<http://www.pureftpd.org/project/pure-ftpd>).
- *wu-ftpd* (<http://www.landfield.com/wu-ftpd/>).

■ Sistemas Windows

- Servidor FTP incluido en *Internet Information Server* (<http://www.iis.net/>).
- *Filezilla Server* (<http://filezilla-project.org/>).
- *Serv-U* (<http://www.serv-u.com/>).

4.2.4. Clientes FTP

Programas que acceden al sistema de ficheros del equipo donde están instalados y establecen conexiones con los servidores FTP para subir o descargar archivos.

Existen múltiples clientes FTP tanto para sistemas libres como para sistemas propietarios. Se pueden clasificar según el interfaz de usuario que ofrecen.

- ◊ **Actividad 4.2:** Consulta el siguiente enlace de *Wikipedia* http://en.wikipedia.org/wiki/Comparison_of_FTP_client_software en el que muestra una comparativa de múltiples clientes FTP.

4.2.4.1. Clientes en línea de comandos

La mayoría de los sistemas operativos integran un cliente que se puede invocar desde la línea de comandos con la orden *ftp*. Para iniciar una conexión se emplea la sintaxis *ftp servidor* (como probaste en la Actividad 4.1).

Una vez establecida la conexión el cliente pone a disposición del usuario una serie de comandos (por ejemplo, *ls*, *get*, *put*, *mget*, *mput*, *cd*, *lcd*...) para listar el contenido de los directorios del servidor, iniciar bajadas y subidas de ficheros, etc. Dependiendo del sistema operativo pueden variar los comandos que podemos utilizar. Con el comando *help* o *?* podemos consultar los comandos disponibles.

Si se quiere ejecutar un comando del equipo local se precede del símbolo *!*, por ejemplo *ls* muestra un listado del directorio actual del servidor y *!ls* o *!dir* muestran un listado del directorio actual del cliente. Hay una excepción, el comando *cd*, para ejecutar el comando *cd* en local hay que usar *lcd* y no *!cd*.

- ◊ **Actividad 4.3:**

- Busca en Internet información sobre los comandos *ftp*. En *Wikipedia* puedes encontrar una lista explicando su función.
- Inicia sesión en **ubuntuXX**. Abre un terminal y ejecuta el comando *ftp*. Introduce *?* para obtener una lista de comandos disponibles. Introduce *help get* para obtener una descripción de la funcionalidad del comando *get*.

4.2.4.2. Clientes “gráficos”

Ofrecen al usuario un interfaz gráfico que facilita la conexión al servidor y la transferencia de ficheros. Suele integrar múltiples funciones adicionales.

Algunos de los más utilizados son:

- *Filezilla client* (<http://filezilla-project.org/>).
- *WinSCP* (<http://winscp.net>).
- *Gftp* (<http://www.gftp.org/>).
- *SmartFTP* (<http://www.smartftp.com/>).
- *CuteFTP* (http://www.globalscape.com/products/ftp_clients.aspx).

4.2.4.3. Navegadores/exploradores

Los navegadores (*Firefox*, *Internet Explorer*, *Google Chrome*, *Safari*,...) y los exploradores de archivos (*Explorer*, *Nautilus*,...) actuales pueden actuar como clientes *ftp*. Para utilizarlos hay que indicar en la dirección que se realizará la conexión a un servidor FTP (como hiciste en la Actividad 4.1).

- Formato general: `ftp://[usuario] [:password]@servidor`.
- Ejemplos: `ftp://ftp.rediris.es`, `ftp://alumno@192.168.100.100`.
- Para realizar conexiones con el usuario **anonymous** no hay que indicar ni usuario ni contraseña.
- Usan el modo activo (explicado posteriormente) por defecto.

Ofrecen un cliente FTP limitado pero sencillo de usar. Permiten instalar complementos adicionales que incluyen clientes FTP más completos.

4.2.5. Protocolo FTP

El protocolo FTP determina el conjunto de normas y reglas en función de las cuales “dialogan” los clientes y los servidores FTP. La comunicación se basa en el envío de mensajes de texto que contienen comandos y respuestas. Utiliza TCP como protocolo de transporte.

- Los comandos FTP son cadenas de caracteres que finalizan con el código de final de línea (`<CR> + <LF>`, retorno de carro seguido del carácter salto de línea).
- Las respuestas FTP son enviadas por el servidor como consecuencia de la acción ejecutada al recibir un comando. Están compuestas por un código de 3 dígitos, que indica cómo se ha procesado el comando enviado, y un mensaje de texto descriptivo. Los dígitos determinan el tipo de respuesta.
 - El primer dígito indica si la acción solicitada por el comando fue exitosa o fallida.
 - El segundo dígito indica a qué se refiere la respuesta.
 - El tercer dígito ofrece información más específica relacionada con el segundo dígito.

◦ **Actividad 4.4:** Consulta la web <http://www.networksorcery.com/enp/protocol/ftp.htm> y observa los comandos y las respuestas del protocolo FTP. En la web hay enlaces a las RFCs que definen el protocolo FTP.

◦ **Actividad 4.5:**

- Inicia una sesión en w7XX con un usuario con privilegios de administrador. Inicia una captura del tráfico de red con Wireshark en modo “no promiscuo”, abre un terminal y utiliza el cliente ftp en línea de comandos para establecer una conexión como usuario anónimo a <ftp.rediris.es>.
- Para la captura en Wireshark, selecciona una trama que contenga el protocolo FTP, haz clic con el botón derecho del ratón y selecciona *Follow TCP Stream*.
- Observa y analiza los comandos y respuestas FTP ¿Qué modo ha usado el cliente FTP para conectarse?

4.2.6. Tipos de acceso

Los servidores FTP permiten, dependiendo de cómo se configuren, dos tipos de acceso desde los clientes.

- Acceso anónimo
 - El cliente FTP se conecta al servidor con un usuario especial anónimo. Como nombres para este usuario se emplean de forma estándar **anonymous** y/o **ftp**.
 - De manera habitual, (no siendo un requisito ya que el administrador del servidor FTP puede decidir otra cosa) el usuario anónimo solo puede descargar archivos y su acceso se limita a un directorio del servidor.
- Acceso autorizado
 - El cliente FTP se conecta con un usuario que debe existir en el servidor. Los usuarios pueden ser:
 - Usuarios locales del sistema operativo donde está instalado el servidor FTP.
 - Usuarios “virtuales”, creados para el acceso FTP. Sus credenciales se pueden almacenar en bases de datos (por ejemplo MySQL), servicios de directorio (LDAP), ficheros de texto, etc.
 - Una vez que se ha autenticado, el usuario accede a un directorio del servidor en el que puede estar o no confinado o “enjaulado” (no puede “subir” a directorios superiores).
 - En el servidor se configuran los privilegios que tiene cada usuario (descargar, subir, borrar, limitación de espacio, acceso a unos directorios u otros, limitación de velocidades, etc.)

4.2.7. Conexiones y modos

FTP es un servicio basado exclusivamente en TCP que utiliza varias conexiones y puertos.

4.2.7.1. Conexión de control y conexiones de datos

Los servidores y los clientes mantienen conexiones TCP independientes para control y transferencia de datos, Figura 4.2.

- Conexión de control.** Inicialmente el cliente establece una conexión con el servidor para “dialogar” con él. Le envía comandos de descarga (*get*), subida (*put*), listado (*ls*), etc. y recibe respuestas del servidor que le informan de cómo atiende las peticiones. Esta conexión permanece activa hasta que el usuario cierra la sesión o hasta que el servidor la finalice porque caduca el tiempo de espera a causa de la inactividad (*timeout*). Los servidores pueden atender múltiples conexiones de control simultáneamente, tantas como se configuren en el servidor para evitar su sobrecarga.
- Conexiones de datos.** Cuando el cliente solicita una transferencia de información se crea una nueva conexión (conexión de datos) que se cierra al finalizar la transmisión. Asociadas a una conexión de control pueden existir múltiples conexiones de datos simultáneas, tantas como transferencias simultáneas y hasta un máximo que se configure en el servidor para evitar su sobrecarga.

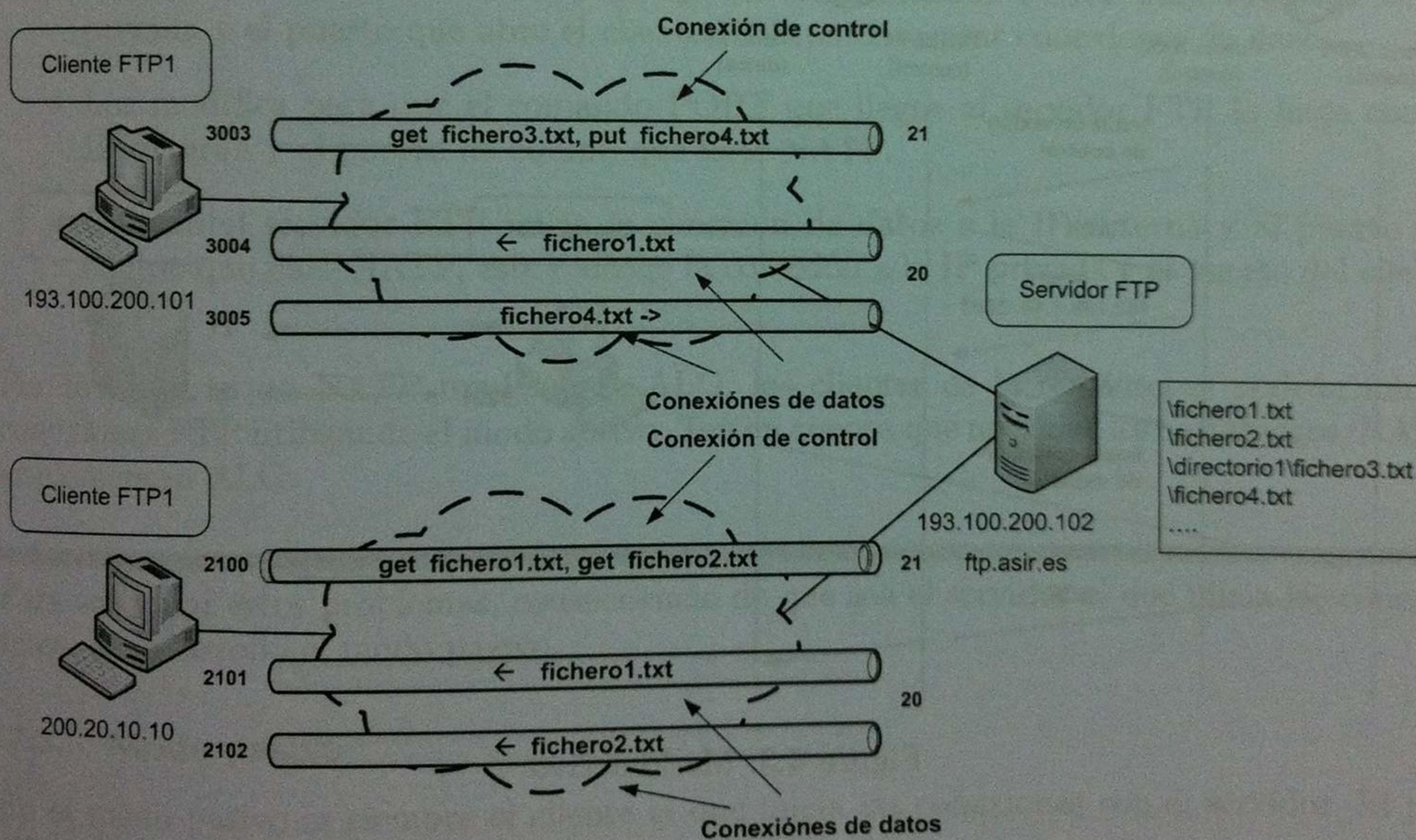


Figura 4.2: Conexiones FTP

Por la conexión de control nunca se envían datos (incluidos listados de directorios) y por las conexiones de datos nunca se envían comandos de control.

Inicialmente los servidores FTP usaban el puerto 21/TCP para atender las conexiones de control (esto continua siendo así) y el puerto 20/TCP para iniciar las conexiones de datos. Actualmente, debido a una serie de problemas que se explican en los siguientes apartados, no usan siempre el

puerto 20 para las conexiones de datos. Los clientes utilizan puertos mayores a 1023 (en el ejemplo 3003, 2100,...) para iniciar o atender conexiones.

Un cliente FTP puede iniciar una conexión a un servidor de dos formas distintas que se conocen como **modo activo** y **modo pasivo**.

¿Sabías que ... ? Es importante conocer el funcionamiento de los modos activo y pasivo para configurar adecuadamente los encaminadores/NATP y los cortafuegos que protegen los servidores FTP. También es importante saber el modo que hay que usar cuando utilizamos un cliente FTP.

4.2.7.2. Modo activo

Es el modo nativo del servicio FTP y su funcionamiento es el mostrado en la Figura 4.3.

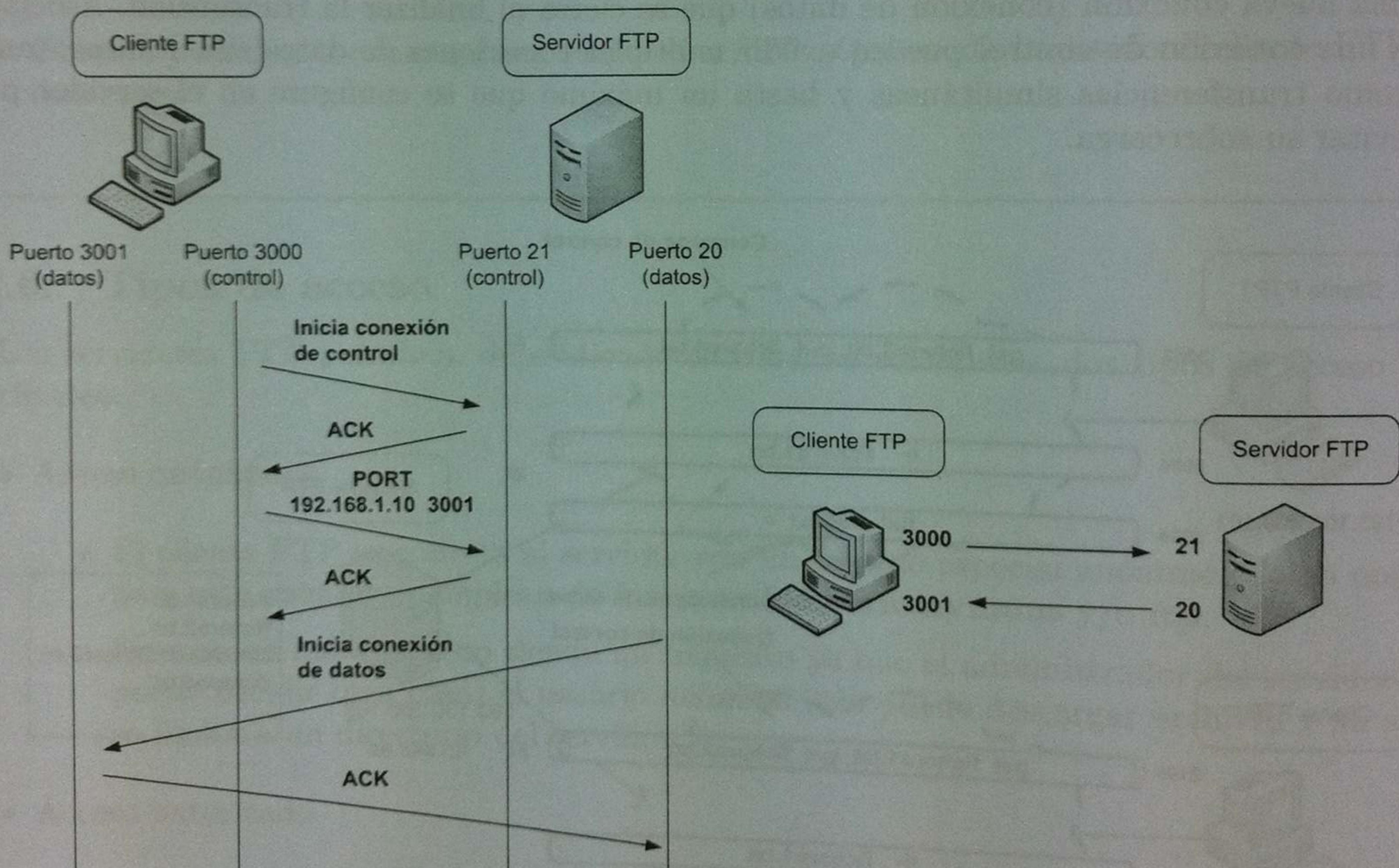


Figura 4.3: Modo activo

1. Se inicia el cliente y establece una conexión de control
 - Abre un puerto local superior a 1023 (en el ejemplo 3000).
 - Establece una conexión TCP con el puerto 21 del servidor.
2. Cuando se solicita una transferencia de ficheros
 - El cliente envía el comando PORT al servidor en el que especifica su dirección IP y un número de puerto que “abrirá” para usar en la conexión de datos (en el ejemplo el 3001).

- El servidor inicia una conexión TCP desde su puerto 20 hacia un puerto que le ha indicado el cliente (en el ejemplo el 3001).
- Se utiliza la conexión de datos para realizar la transferencia de información.

En este modo, es el servidor el que inicia las conexiones de datos y el cliente tiene que abrir puertos para atender dichas conexiones. La máquina que ejecuta el cliente FTP tiene que aceptar conexiones a puertos, usados en las transferencias de datos, superiores a 1023. Esto puede comprometer la seguridad del equipo.

- Los cortafuegos instalados en el equipo donde se encuentra el cliente FTP o en la red a la que pertenece evitarán estas conexiones aleatorias para prevenir ataques.
- Si el equipo donde está el cliente está detrás de un encaminador/NATP este descartará las conexiones iniciadas desde el exterior por el Servidor FTP a los puertos que abre el cliente. Es muy habitual que los equipos de una red privada que se conecta a Internet lo hagan a través de un encaminador que implementa NATP. Recomendamos consultar el apartado dedicado a NAT del Tema 1 para entender este escenario.

¿Sabías que ... ? Los cortafuegos actuales y versiones modernas de NATP implementan FTP ALG (*Application Level Gateway*). Es un mecanismo que:

- Analiza las conexiones FTP y se fija en los comandos PORT para registrar la IP privada y el puerto que abre el cliente FTP para aceptar conexiones de datos.
- Los modifica para que el comando PORT que llegue al servidor FTP lo haga con la IP externa y el puerto de equipo que hace NATP.
- Cuando el servidor FTP inicia la conexión de datos a la IP externa y al puerto del equipo que hace NATP, este redirige la conexión a la IP privada y al puerto del cliente FTP.

Por lo tanto, si un NATP implementa ALG, los clientes de la red interna podrán iniciar conexiones FTP utilizando el modo activo. Ten en cuenta que no todos los cortafuegos/NATP implementan ALG.

Para solventar estos problemas, consecuencia de que sea el servidor el que inicia las conexiones de datos, se desarrolló el modo pasivo.

4.2.7.3. Modo pasivo

En el modo pasivo es siempre el cliente el que inicia las conexiones con el servidor. El puerto 20 del servidor no se utiliza, véase Figura 4.4.

1. El cliente se inicia y establece una conexión de control
 - Abre un puerto local superior a 1023 (en el ejemplo 3000).
 - Establece una conexión TCP con el puerto 21 del servidor.
 - Esto es igual que el modo activo.
2. Cuando se solicita una transferencia de ficheros

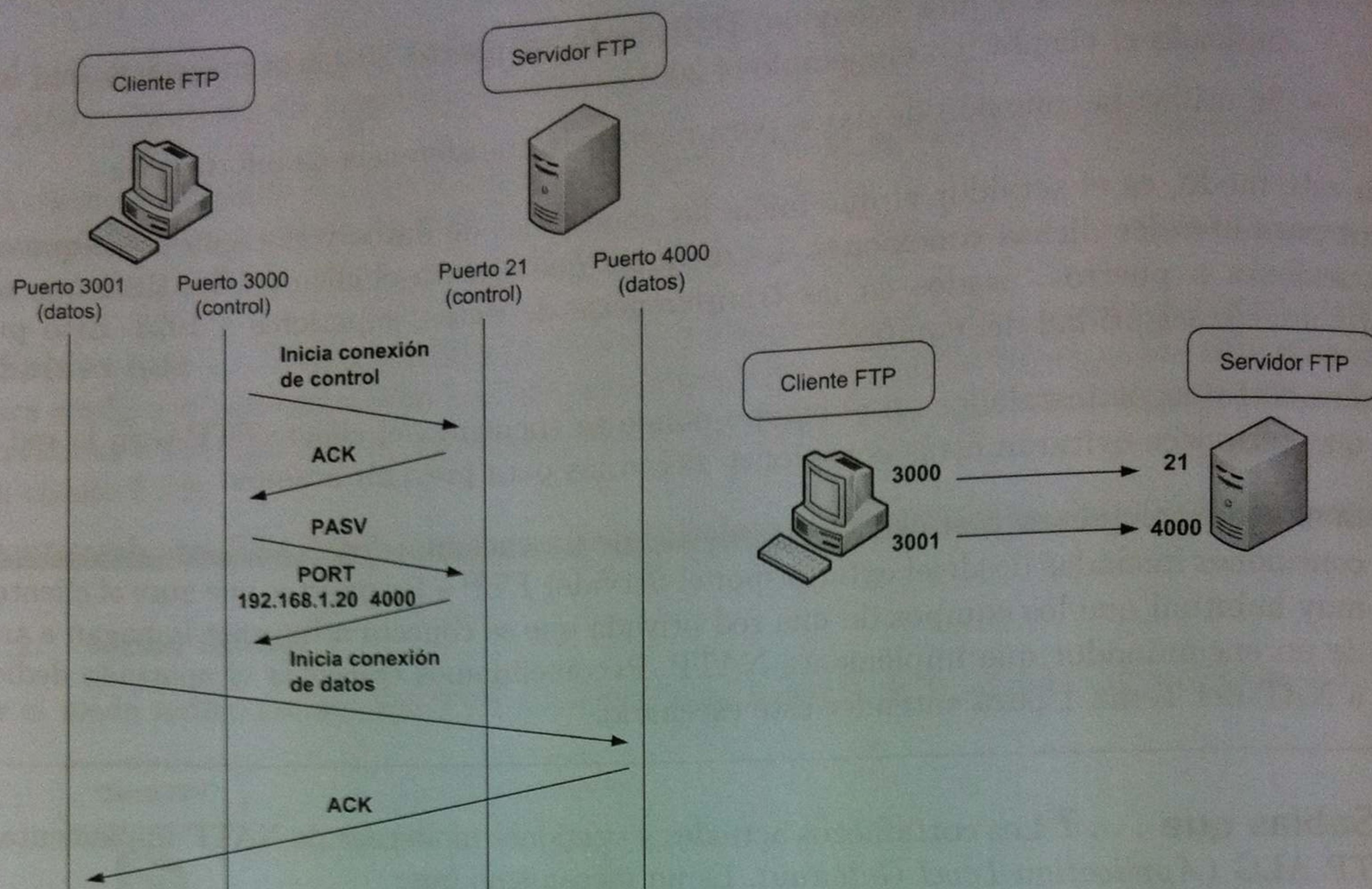


Figura 4.4: Modo pasivo

- El cliente envía el comando PASV para activar el modo pasivo. Como respuesta a este comando, el servidor retorna un número de puerto que tenga disponible (en el ejemplo 4000).
- El cliente inicia una conexión TCP, abre un puerto local superior al 1023 (en el ejemplo 3001) hacia el puerto que le envió el servidor (en el ejemplo 4000).
- Se utiliza la conexión de datos para realizar la transferencia de información.

El modo pasivo resuelve el problema de que el cliente tenga que aceptar conexiones en puertos mayores a 1023 pero lo traslada al servidor.

- La máquina donde se ejecuta el servidor FTP tiene que aceptar conexiones en múltiples puertos y esto es una amenaza para la seguridad del equipo. Los cortafuegos actuales permiten realizar un seguimiento de las conexiones pasivas de datos, comprobando que el cliente que solicita la conexión al puerto especificado por el servidor se corresponde con el cliente al que se le indicó ese puerto y que por tanto la conexión se establece para el envío o recepción de datos.
- Si el servidor está detrás de un NATP hay que
 - Configurar en el servidor la IP externa que usa el NATP y un rango de puertos para aceptar conexiones de datos.
 - Redirigir el rango de puertos del encaminador que hace NATP al equipo donde está el servidor FTP.

4.2.7.4. Cortafuegos y encaminadores/NATP

El uso de servidores y clientes FTP y los modos activo y pasivo implican una configuración adecuada de los cortafuegos y de los encaminadores/NATP que existan en los equipos y redes donde se utilizan. Resumimos a continuación algunas situaciones comunes, Figura 4.5.

■ Cliente FTP

- Conexión en modo activo

- El cortafuegos instalado en el equipo cliente tiene que permitir conexiones TCP salientes hacia el puerto 21 y conexiones TCP entrantes a puertos mayores que 1023 desde el puerto 20.
- Si el cliente está detrás del cortafuegos de red y/o un encaminador/NATP
 - ◊ Si el NATP no implementa FTP ALG. Existirán problemas porque se filtrarán las conexiones TCP iniciadas desde el exterior por los servidores FTP a puertos mayores que 1023 del cliente.
 - ◊ Si el NATP implementa FTP ALG. Los clientes podrán usar el modo activo porque se permitirán conexiones entrantes a puertos mayores que 1023.

- Conexión en modo pasivo. Los cortafuegos existentes tienen que permitir conexiones TCP salientes hacia el puerto 21 y hacia puertos mayores que 1023.

■ Servidor FTP

- Acepta conexiones en modo activo

- Los cortafuegos existentes tienen que permitir conexiones TCP entrantes al puerto 21 del servidor.
- Los cortafuegos existentes tienen que permitir conexiones TCP salientes desde el puerto 20 del servidor hacia puertos mayores que 1023.

- Acepta conexiones en modo pasivo

- El cortafuegos instalado en el servidor tiene que permitir conexiones TCP entrantes hacia el puerto 21 y conexiones TCP entrantes a puertos mayores que 1023. Es recomendable que el cortafuegos haga un seguimiento de las conexiones pasivas de datos, comprobando que el cliente que solicita la conexión al puerto especificado por el servidor se corresponde con el cliente al que se le indicó ese puerto.
- Si el servidor está detrás de un cortafuegos de red y/o encaminadores/NATP
 - ◊ Los cortafuegos de red tienen que permitir las conexiones TCP entrantes al puerto 21.
 - ◊ Hay que redirigir el puerto 21 del encaminador/NATP al puerto 21 del servidor.
 - ◊ Se tiene que configurar en el servidor la IP externa del NATP y un rango de puertos para aceptar conexiones de datos.
 - ◊ Los cortafuegos de red deben permitir las conexiones TCP entrantes hacia los puertos definidos en el rango.
 - ◊ Hay que redirigir el rango de puertos del encaminador/NATP al servidor FTP.

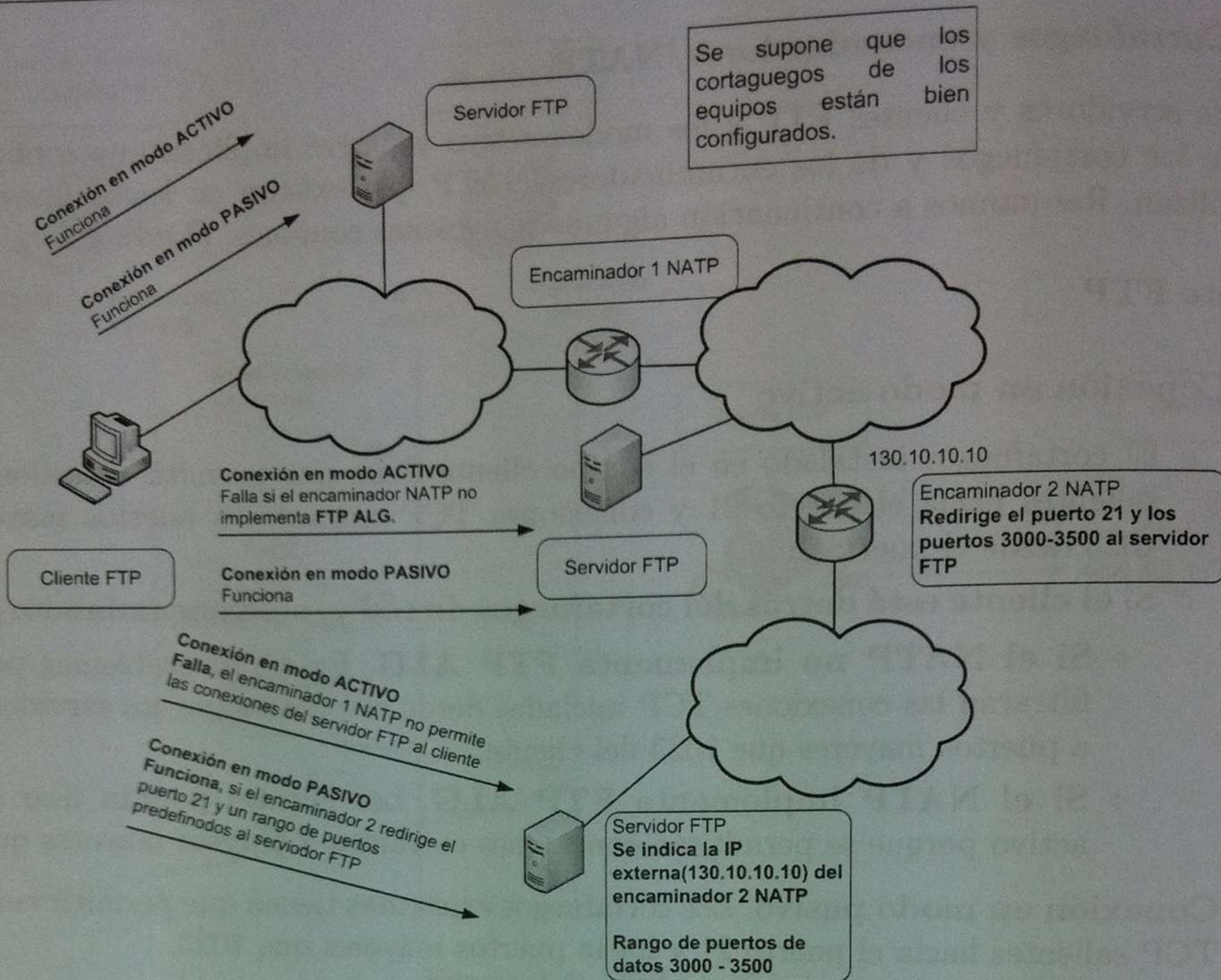


Figura 4.5: Conexiones en modo activo y pasivo

4.2.7.5. Resumen y comparativa de modos

▪ Modo activo

- Conexión de control: Cliente ($>1023 \Rightarrow 21$)
- Conexiones de datos: Cliente ($>1023 \Leftarrow 20$)

▪ Modo pasivo

- Conexión de control: Cliente ($>1023 \Rightarrow 21$)
- Conexiones de datos: Cliente ($>1023 \Rightarrow >1023$)

El modo activo facilita la configuración y la administración del servidor FTP pero presenta problemas de seguridad a los clientes y problemas de acceso si están detrás de un cortafuegos y/o encaminador NATP. El modo pasivo favorece al cliente pero implica una configuración más compleja en el servidor.

4.2.8. Tipos de transferencia de archivos

En FTP existen dos modos de transferencia de archivos, ASCII y binario:

- **Formato ASCII (*type ascii*)**. Se transmite *byte* a *byte*. Para archivos de texto (*txt, html, java,...*)
- **Formato binario (*type bin*)**. Se transmite *bit* a *bit*. Para archivos que no son de texto (ejecutables, imágenes, videos,...)

Los clientes FTP permiten definir el formato de transmisión (*ascii* o *bin*) a utilizar en función del tipo de archivo a transferir. Algunos clientes, actualmente casi todos, ofrecen modo automático que detecta el tipo de archivo y establece el tipo de transferencia adecuado.

4.2.9. Seguridad

FTP no es un protocolo seguro. Fue diseñado para ofrecer velocidad pero no seguridad. Se utilizan mecanismos de autenticación de usuarios para determinar los privilegios de acceso y transferencia en el servidor, pero:

- No se usan mecanismos para garantizar que los equipos involucrados en la transferencia son quienes dicen ser. Es vulnerable a ataques de suplantación de identidad (*spoofing*).
- Todo el intercambio de información, incluyendo el usuario y *password* y la transferencia de cualquier archivo, se realiza en “texto plano” sin ningún tipo de cifrado. Es vulnerable a ataques de análisis de tráfico de red (*sniffing*).

Los clientes y servidores FTP pueden tener vulnerabilidades y ser aprovechadas por potenciales atacantes para comprometer los datos y los equipos donde se ejecutan.

¿Sabías que ... ? La mayoría de los protocolos TCP/IP (FTP, HTTP, SMTP, Telnet, POP, IMAP, DNS,...) no son seguros porque en su diseño inicial no se pensó en la seguridad.

4.2.10. FTPS (o FTP/SSL)

Conjunto de especificaciones que determinan cómo encapsular FTP en SSL (*Secure Sockets Layer*) o en TLS (*Transport Layer Security*) para ofrecer comunicaciones seguras. Gracias a la utilización de algoritmos criptográficos y certificados digitales se puede garantizar la confidencialidad y la integridad de la información transmitida, así como la autenticidad de los servidores.

Existen dos métodos para implementar FTPS, FTPS Explícito (FTPES) y FTPS Implícito:

▪ FTPS Implícito

- El cliente establece una conexión de control y se establece la conexión SSL/TLS.
- Si el servidor no soporta FTPS se cierra la conexión.
- Todas las comunicaciones, conexión de control y conexiones de datos, son cifradas. El cliente y el servidor no negocian.
- Para mantener la compatibilidad con los clientes FTP que no soporten SSL/TLS se utilizan otros puertos para atender las peticiones FTPS (se usan como puertos estándar el 990/TCP para control y el 989/TCP para datos).

▪ FTPS Explícito (FTPES)

- El cliente establece una conexión de control al puerto 21, solicita explícitamente que la comunicación sea segura enviando el comando AUTH SSL o AUTH TTL y si el servidor lo soporta se establece una conexión SSL/TLS basándose en algoritmos criptográficos y certificados digitales.

- Si el servidor no soporta FTPS le ofrece al cliente la posibilidad de usar FTP "normal" no seguro.
- El cliente y el servidor pueden negociar qué parte de las comunicaciones, conexión de control y/o conexiones de datos serán cifradas.
- Es el método recomendado porque permite mayor control sobre la comunicación.

No hay que confundir **FTPS** con **SFTP** (*SSH File Transfer Protocol*) el protocolo de transferencia de ficheros basado en SSH que tratamos en apartados posteriores. Ni con enviar el protocolo FTP a través de una conexión SSH (túnel FTP sobre SSH) conocido como **Secure FTP**. Por lo tanto **FTPS**, **SFTP** y **Secure FTP** son diferentes.

4.2.11. Protocolo FXP

FXP (*File eXchange Protocol*) es un protocolo de transferencia de datos directa entre servidores FTP, utilizando un cliente solo para conectarlos inicialmente. Esto significa que el ancho de banda del cliente es solo para la conexión inicial y no para la transferencia de ficheros que se hace directamente de un servidor a otro, Figura 4.6. Para que sea posible los servidores FTP tienen que permitirlo.

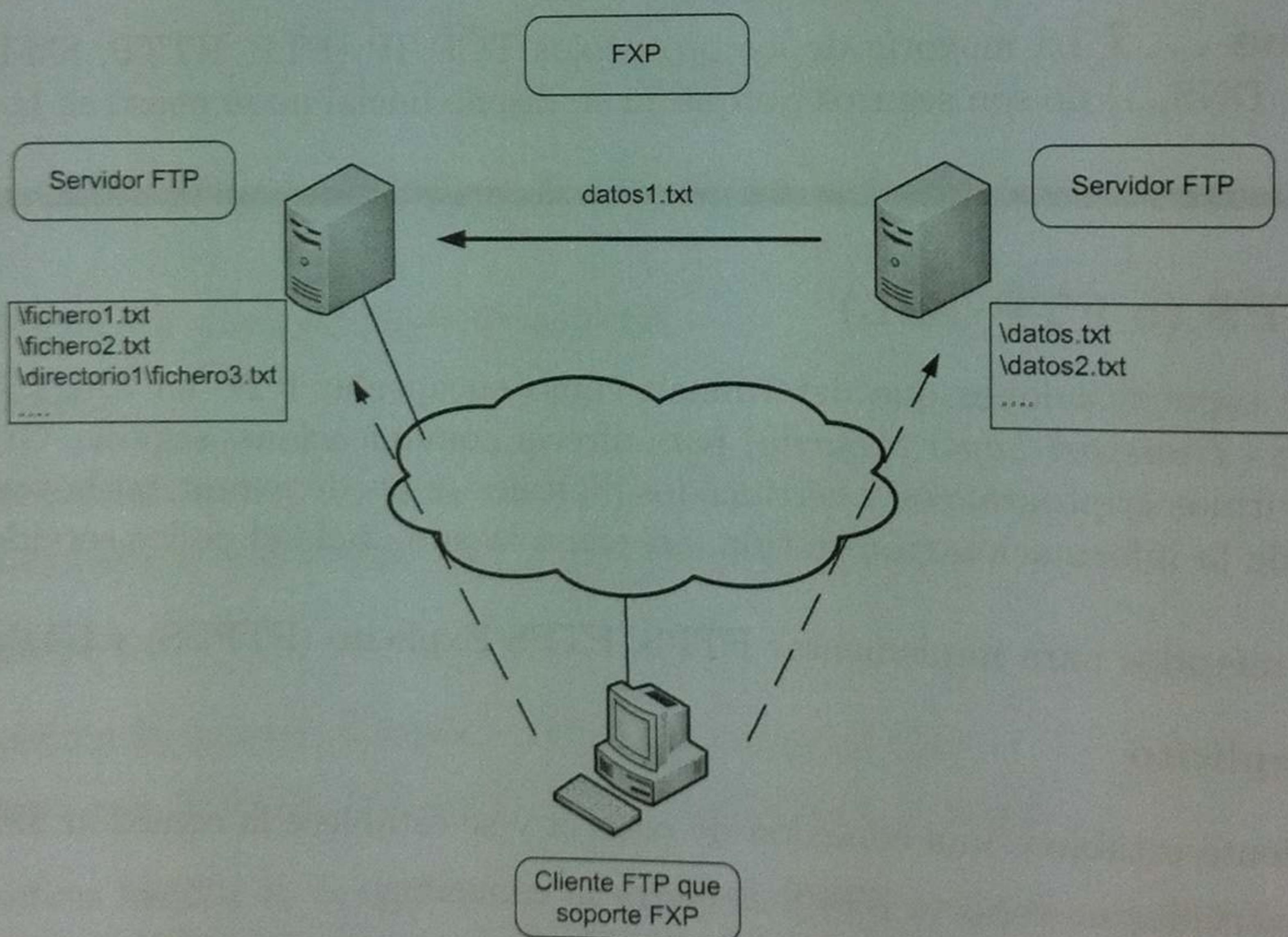


Figura 4.6: FXP

FTPX se puede utilizar, por ejemplo, si quieras migrar ficheros de un servidor FTP a otro ahorrando la descarga desde un servidor al cliente y posteriormente la subida del cliente al otro servidor (se consigue mayor rapidez y menos sobrecarga de la red).

4.3. Servicio TFTP

TFTP (*Trivial File Transfer Protocol*) es un protocolo de capa de aplicación diseñado para ofrecer un servicio de transferencia de ficheros simple y rápido basado en el modelo cliente/servidor.

Al igual que en FTP existen clientes TFTP y servidores TFTP.

Sus características principales son:

- TFTP utiliza UDP como protocolo de nivel de transporte. Los servidores TFTP usan el puerto 69/UDP como puerto estándar.
- No existen mecanismos de autenticación o cifrado.

Al utilizar el protocolo UDP la capa de transporte no garantiza la integridad de la información transmitida pero la velocidad de transferencia es mayor que en FTP.

Se utiliza principalmente en estaciones o dispositivos de red para cargar y hacer copias de seguridad del sistema operativo, archivos de configuración, aplicaciones, etc.

¿Sabías que ...? CISCO, una de los principales fabricantes de *routers* y *switches*, usa TFTP para realizar copias de seguridad de archivos de configuración y cargar nuevas versiones del sistema operativo en sus dispositivos.

4.4. Servicios SFTP/SCP

SSH (*Secure Shell Protocol*) es un protocolo de capa de aplicación diseñado para ofrecer un servicio de acceso a terminales de equipos remotos. Está basado en el modelo cliente/servidor. El cliente SSH permite establecer conexiones a terminales del equipo donde ejecuta el servidor SSH. Los servidores SSH usan el puerto 22/TCP como puerto estándar.

¿Sabías que ...? SSH es uno de los protocolos de acceso remoto más usados. Aunque no se trata con profundidad en el libro recomendamos ampliar información sobre su funcionalidad y utilización.

SSH, a diferencia de otros protocolos de acceso remoto, como por ejemplo *Telnet*, ofrece autenticación, confidencialidad e integridad.

- Se autentifica a los dos extremos de la conexión
 - El servidor se autentica ante el cliente con una clave.
 - El cliente se autentica ante el servidor.
- Se cifran los datos intercambiados
 - Nombres de usuarios y *passwords* viajan cifrados.
 - La información transmitida viaja también cifrada.

SSH, además de otras funcionalidades, integra mecanismos de transferencia de ficheros garantizado igualmente autenticación, confidencialidad e integridad. Se basa en los protocolos SFTP (*SSH File Transfer Protocol*) y SCP (*Secure Copy Protocol*).

- SFTP

- Permite la transferencia de ficheros entre sistemas remotos.
- Permite listar ficheros y directorios del servidor.
- Permite realizar funciones adicionales en el servidor como renombrar, borrar, crear archivos y carpetas, cambiar permisos, descomprimir,...

- SCP

- Permite la “copia” de ficheros entre sistemas remotos.
- Hay clientes SCP gráficos que integran funcionalidades adicionales como listar, borrar, etc. No son clientes scp “puros”.

Los **servidores SSH** atienden peticiones de transferencia de ficheros desde clientes SFTP y/o SCP. Ejemplos de servidores SSH son *OpenSSH* (<http://www.openssh.com/>) y *WinSSHD* (<http://www.bitvise.com/winsshd>).

Existen múltiples **clientes SFTP/SCP** y se pueden clasificar según el interfaz de usuario que ofrecen:

- **Clientes en línea de comandos**

- Clientes que se pueden invocar desde la línea de comandos.
- Clientes scp y sftp (este último ofrece comandos similares a los clientes ftp como *get*, *put*, *mget*, *mput*,...).

- **Clientes gráficos**

- La mayoría de los clientes gráficos FTP también pueden actuar como clientes SFTP/SCP.

4.5. Prácticas resueltas

Práctica 4.1: Clientes FTP

Los clientes FTP se pueden clasificar según el interfaz de usuario que ofrecen. En esta práctica se prueban varios clientes FTP para descargar archivos desde un servidor FTP público.

1. Cliente ftp en línea de comandos

- 1.1. Inicia sesión en **debianrouterXX** o **ipcopXX** con el usuario **alumno** (si no existen el usuario alumno inicia sesión como **root** y crea la cuenta ejecutando *adduser alumno*).
- 1.2. Crea un directorio denominado **pruebasFTP** dentro de la carpeta *home* del usuario **alumno** y cámbiate al directorio creado.

mkdir pruebasFTP

cd pruebasFTP

- 1.3. Dentro del directorio **pruebasFTP** crea un fichero de texto denominado **datos1.txt** con el contenido que quieras.
- 1.4. Establece una conexión como usuario **anonymous** al servidor **ftp.rediris.es** con el cliente ftp en línea de comandos.

ftp ftp.rediris.es

- a. Introduce **anonymous** como usuario.
- b. Deja la contraseña en blanco

- 1.5. Ejecuta el comando **?** para mostrar los comandos ftp disponibles.

ftp >?

- 1.6. Ejecuta el comando **pwd** para ver la ruta en la que te encuentras en el servidor.

ftp > pwd

- 1.7. Ejecuta el comando **!pwd** para ver la ruta en la que te encuentras en el cliente (debe ser **/home/alumno/pruebasFTP**) (recuerda que los comandos que ejecutes con el símbolo **!** delante se ejecutan en el cliente, excepto el comando **cd** que para ejecutarlo en local hay que usar **lcd** y no **!cd**).

ftp >!pwd

- 1.8. Ejecuta el comando **!ls** para ver un listado del directorio donde estas en el cliente.

ftp > ls

- 1.9. Ejecuta el comando `ls` para ver un listado del servidor FTP.

`ftp > ls`

- 1.10. Descarga el fichero `welcome.msg` usando el comando `get`.

`ftp > get welcome.msg`

- 1.11. Ejecuta el comando `!ls` para ver un listado del directorio donde estás en el cliente y comprobar que se ha descargado el fichero.

`ftp >!ls`

- 1.12. Desde el cliente ftp crea un directorio denominado **imagenes** usando `!mkdir` dentro del directorio **pruebasFTP** y cámbiate a él.

`ftp >!mkdir imagenes`

`ftp > lcd imagenes`

- 1.13. Vuelve al directorio **pruebasFTP** y sube el fichero **datos1.txt** al servidor usando el comando `put`.

`ftp > lcd..`

`ftp > put datos1.txt`

a. ¿Se ha subido el fichero al servidor? ¿Por qué?

- 1.14. Cierra la conexión ftp con el comando `bye`.

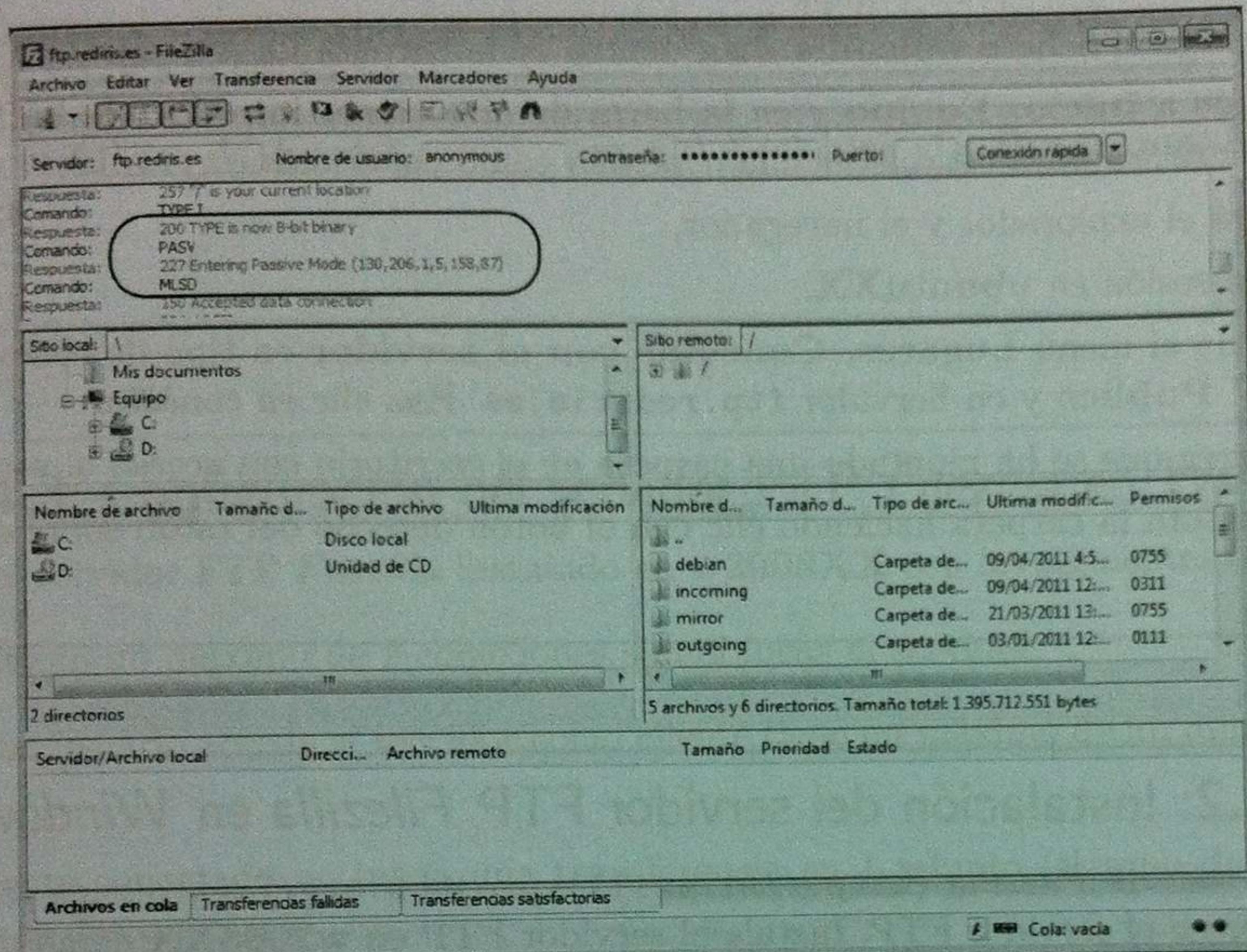
`ftp > bye`

2. Cliente ftp gráfico *Filezilla*

Filezilla (<http://filezilla-project.org/>) es una solución FTP libre que pone a disposición de los usuarios un cliente FTP multiplataforma y un servidor FTP para sistemas *Windows*. Ambos son distribuidos bajo licencia GNU (*General Public License*). En esta práctica vamos a instalar y utilizar el cliente FTP.

- 2.1. Inicia sesión en **w7XX** con un usuario con privilegios de administrador.
- 2.2. Crear la carpeta **C:\pruebasFTP** y dentro crea un fichero de texto denominado **datos1.txt** con el contenido que quieras.
- 2.3. Accede a la web de *Filezilla* y descarga el cliente FTP.
- 2.4. Instala el cliente con las opciones por defecto.
- 2.5. Inicia *Filezilla* y establece una conexión como usuario **anonymous** al servidor `ftp.rediris.es`:
 - a. En el cuadro de texto **Servidor** introduce `ftp.rediris.es`.

- b. En nombre de usuario puedes introducir **anonymous** o dejarlo en blanco, porque por defecto usa el usuario **anonymous**. Usa por defecto el puerto 21.
- c. Haz clic en **Conexión rápida**.
- d. Una vez establecida la conexión puedes ver en los mensajes del protocolo FTP intercambiados entre el cliente y el servidor en el área superior y los directorios y ficheros del cliente (sitio local) y el servido (sitio remoto) en la parte izquierda y derecha respectivamente, véase Figura 4.7. ¿Qué modo ha usado el cliente para descargar el listado de ficheros del servidor, activo o pasivo?, lo puedes saber observando los mensajes intercambiados.

Figura 4.7: Cliente *Filezilla*

- 2.6. Descarga el fichero **welcome.msg** dentro de la carpeta **C:\PruebasFTP**
 - a. Accede a la carpeta **C:\PruebasFTP** en la parte cliente.
 - b. Haz clic con el botón derecho del ratón sobre el fichero **welcome.msg** y selecciona **descargar** o arrástraloo a la carpeta **pruebasFTP** del directorio local.
- 2.7. Sube el fichero **datos1.txt** al servidor. ¿Se ha subido el fichero al servidor? ¿Por qué?
- 2.8. Observa un resumen de las transferencias fallidas y satisfactorias en la parte inferior de *Filezilla*.
- 2.9. Cierra el cliente FTP.
- 2.10. Inicia sesión en **ubuntuXX** con un usuario con privilegios de administrador.
- 2.11. Instala el cliente FTP *Filezilla* para usarlo en futuras prácticas. Como está incluido en los repositorios oficiales de *Ubuntu* puedes descargarlo e instalarlo desde el **Centro de software (Aplicaciones, Centro de software de Ubuntu)** o desde un terminal ejecutando **sudo apt-get install filezilla**.
- 2.12. Accede al menú **Aplicaciones, Internet. Filezilla** e inicia el programa. Establece una conexión como usuario **anonymous** al servidor **ftp.rediris.es**. Observa que el interfaz es similar a la versión para *Windows*.

- 2.13. Cierra el cliente FTP.

3. Navegadores\exploradores como clientes FTP

- 3.1. Inicia sesión en **w7XX** con un usuario con privilegios de administrador.
- 3.2. Accede a la web de *Firefox* (<http://www.mozilla-europe.org/es/>), descarga la última versión del navegador (la versión 4 en el momento de redactar está práctica) e instala el navegador con las opciones por defecto.
- 3.3. Inicia *Firefox* e introduce la URL `ftp://ftp.rediris.es` para establecer una conexión como usuario **anonymous**.
- 3.4. Descarga el fichero `welcome.msg` dentro de la carpeta **C:\PruebasFTP**.
- 3.5. Accede a **Inicio**, **Equipo** y en la barra de direcciones introduce la URL `ftp://ftp.rediris.es` para establecer una conexión como usuario **anonymous**.
- 3.6. Cierra el explorador y el navegador.
- 3.7. Inicia sesión en **ubuntuXX**.
- 3.8. Accede al menú **Lugares**, **Conectar con el servidor** en tipo de conexión introduce **FTP Publico** y en Servidor `ftp.rediris.es`. Haz clic en conectar.
- 3.9. Observa que se ha montado una carpeta en el escritorio con acceso al servidor FTP.
- 3.10. Desmonta la carpeta haciendo clic con el botón derecho del ratón sobre ella.

Práctica 4.2: Instalación del servidor FTP *Filezilla* en *Windows*

Accede a la web de *Filezilla* (<http://filezilla-project.org/>) e investiga sobre las características que ofrece el servidor FTP. Instala el servidor FTP en **w2008XX** como un servicio que se arranque manualmente. Configura como puerto de administración el 14147 (el que se propone por defecto) e instala la herramienta de administración del servidor.

1. Inicia una sesión en **w2008XX** como **administrador**.
2. Accede a la web de *Filezilla* y descarga el servidor FTP.
3. Inicia la instalación del servidor.
4. Acepta los términos de la licencia.
5. Selecciona el tipo de instalación **Standard**. Observa que instala todos los componentes, incluida la herramienta de administración del servidor, excepto el código fuente. Haz clic en *Next*.
6. Deja la carpeta de instalación por defecto y haz clic en *Next*.
7. Selecciona “*Install as service, started manually*” deja como Puerto de administración el 14147 y haz clic en *Next*. Observa que está marcada la opción para que el servidor se inicie cuando finalice la instalación.
8. Selecciona “*Start manually*” para que la herramienta de administración del servidor no se inicie automáticamente cuando los usuarios inicien sesión. Haz clic en *Install*.

9. Haz clic en *Close* cuando finalice la instalación. Se iniciará automáticamente la herramienta de administración del servidor. Por defecto ofrece conectarse al equipo local (127.0.0.1) y al puerto 14147. Haz clic en **OK** y observa la interfaz que ofrece (la usaremos en la próxima práctica). Ten en cuenta que has dejado la contraseña solicitada en blanco y te has podido conectar.
10. Accede a **Inicio, Herramientas administrativas, Servicios** y comprueba que se ha creado un servicio asociado al servidor *Filezilla* (*Filezilla Server FTP server*) que se inicia manualmente. Observa que está iniciado.
11. Abre un terminal y ejecuta el comando *netstat -a -p TCP -n* y observa que los puertos TCP 21 y 14147 están a la escucha.
12. Consulta el fichero de *log* del servidor en **C:\Archivos de Programa\FileZilla Server\Logs**.

◊

Práctica 4.3: Configuración del servidor FTP *Filezilla* en Windows

Configura el servidor FTP *Filezilla* instalado en **w2008XX** con las siguientes opciones:

- Se permitirán un máximo de 2 conexiones simultáneas al servidor.
- Cuando un usuario accede al servidor como cliente tiene un máximo de un minuto para hacer *login*.
- Si un cliente conectado no inicia una transferencia en 1 minuto (tiempo de inactividad) o menos se le desconectará.
- El mensaje de bienvenida del servidor será “Bienvenido al servidor FTP Filezilla de la red virtual XX”.
- No se permitirán conexiones desde la IP 10.33.XX.2.
- Para administrar el servidor FTP la interfaz de administración empleará el puerto 14147 y se podrá acceder desde el equipo local (donde está instalado *Filezilla Server*). Para acceder a través de la interfaz se deberá indicar la contraseña “garceta”.
- Se habilitará el *log* del servidor con ficheros de 200 KB como máximo. Se usará un fichero de *log* para cada día y se borrarán los ficheros de *logs* con más de 10 días de antigüedad.
- La velocidad de bajada del servidor será de 10 KB/s como máximo de lunes viernes y el fin de semana la velocidad será la máxima permitida. Respecto a la velocidad de subida será de 5 KB/s de lunes a viernes y en fin de semana 15 KB/s.
- Se “banearán” durante 2 horas a las conexiones desde direcciones IP que fallen 10 veces en la autenticación al conectarse al servidor.
- Se crearán los siguientes usuarios:
 - Usuario anónimo (usuario **anonymous** sin contraseña).

- Directorio predeterminado C:\ftp\pub. El usuario solo tendrá permisos de lectura, es decir de descarga de archivos.
- No pertenece a ningún grupo.
- Usuario **profesor** (con contraseña).
 - Directorio predeterminado C:\ftp\. El usuario tendrá permisos de lectura, escritura, borrado y modificación de ficheros y directorios.
 - No pertenece a ningún grupo.
- Usuarios **alumno1** y **alumno2** (con contraseña).
 - Pertenecerán al grupo **alumnos**.
 - El directorio (C:\ftp\alumnos) es el predeterminado para todos los miembros de grupo **alumnos** y todos tendrán permisos de lectura y escritura de ficheros y directorios.
 - La velocidad de subida para todos los alumnos será de 2KB/s.

Conéctate al servidor FTP desde clientes FTP instalados en otros equipos de la red y comprueba que se cumple la configuración realizada.

1. Configuración global del servidor

- 1.1. Inicia sesión en **w2008XX** con un usuario con privilegios de administrador.
- 1.2. Inicia el servidor FTP si no lo está en **Inicio, Todos los programas, Filezilla Server, Start Filezilla Server**.
- 1.3. Accede la herramienta de configuración del servidor desde **Inicio, Todos los programas, Filezilla Server, Filezilla Server Interface**. Conéctate al servidor del equipo local (127.0.0.1), Figura 4.8.

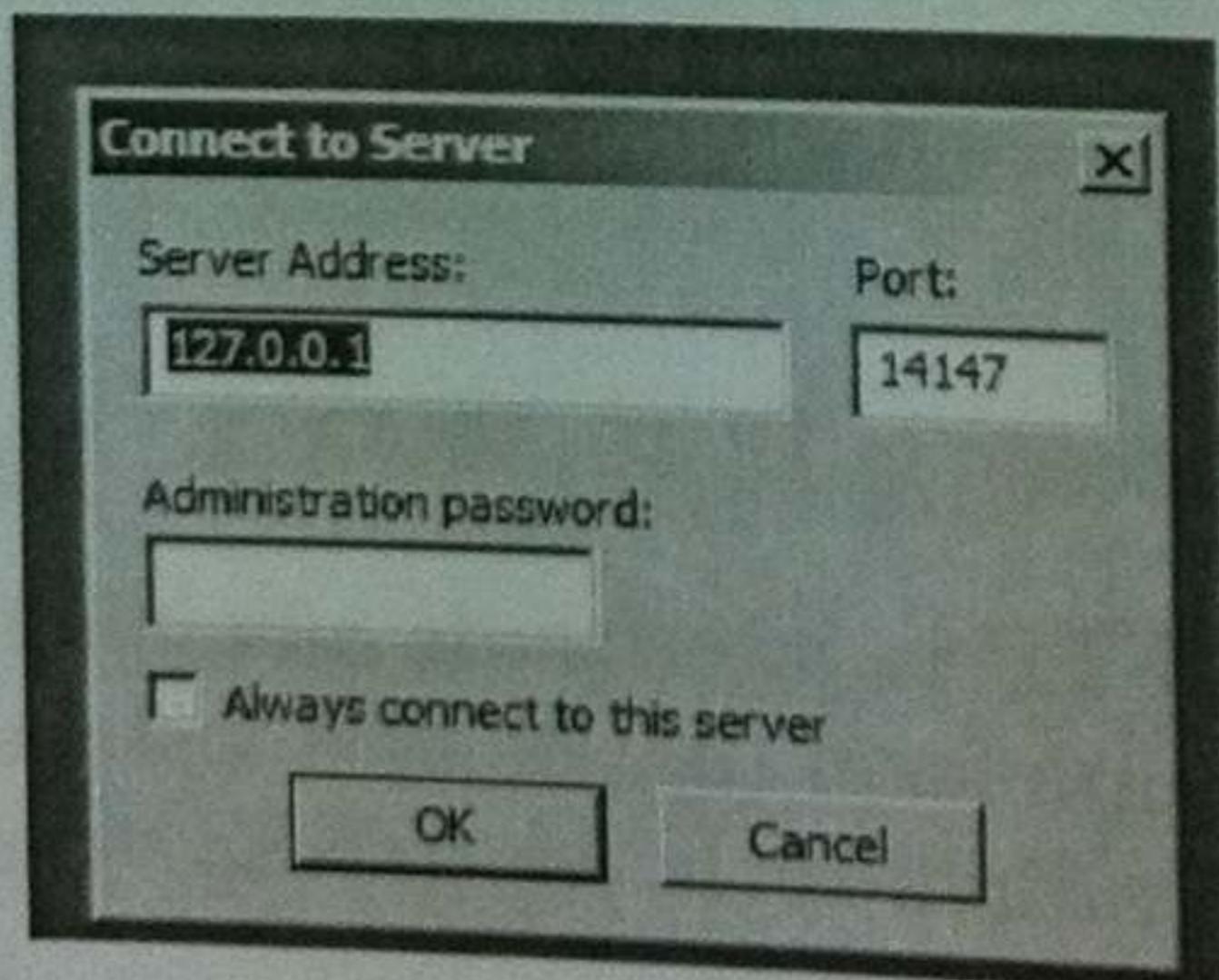


Figura 4.8: Herramienta de administración de Filezilla FTP Server

- 1.4. En la barra de tareas accede a **Edit, Settings** para configurar las opciones globales del servidor.
- 1.5. Navega a través de las opciones para realizar la configuración pedida, véase Figura 4.9.
 - a. En *General settings*, *Max. number of users* introduce 2.
 - b. En *General settings*, *Login Timeout* introduce 60.
 - c. En *General settings*, *No Transfer Timeout* introduce 60.
 - d. En *General settings*, *Welcome message* introduce el mensaje de bienvenida "Bienvenido al servidor FTP de la red virtual XX".

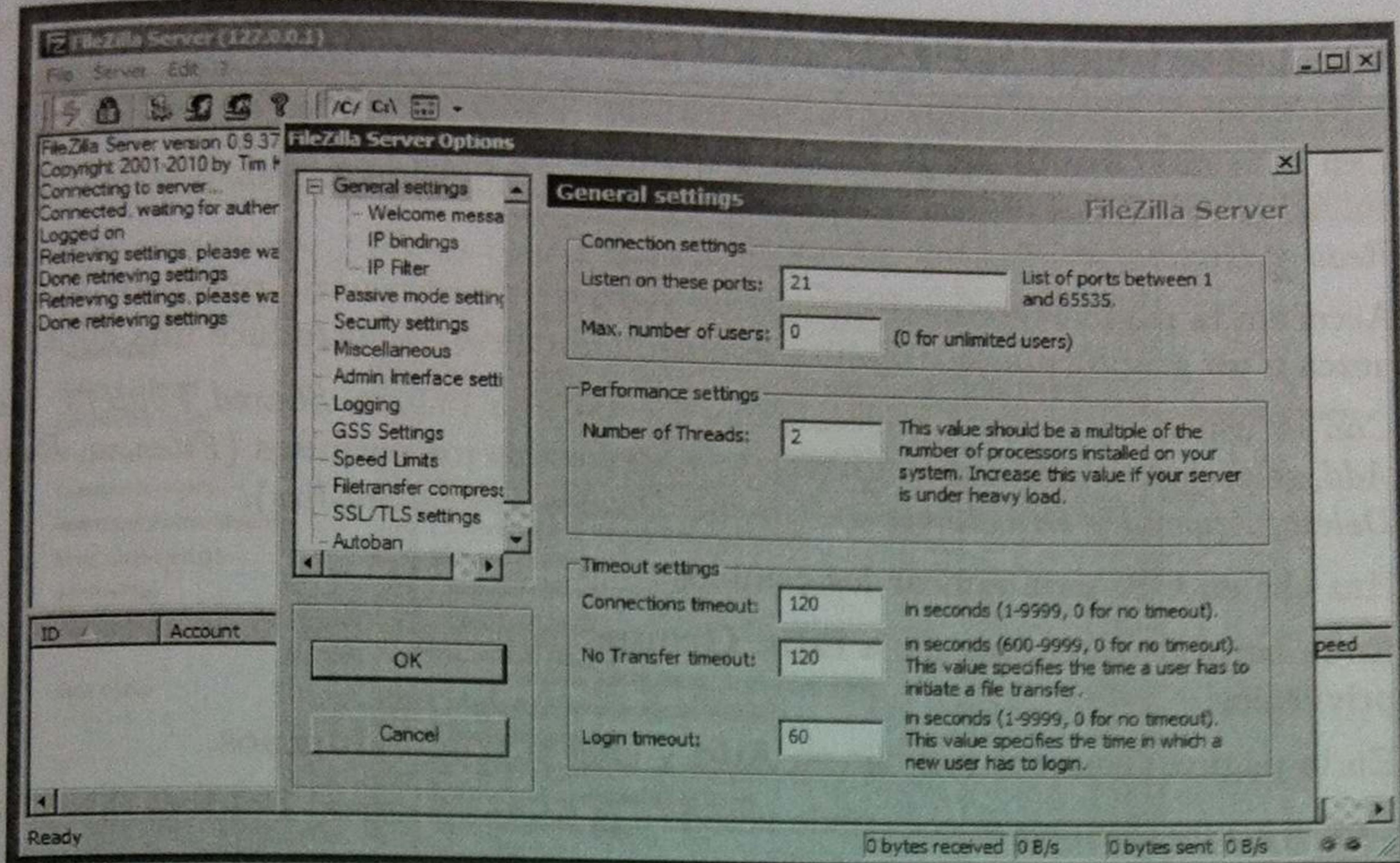


Figura 4.9: Configuración global de Filezilla FTP Server - 1

- e. En *General settings, IP Filter*, *The following IP addresses are no allowed to connect to the server* introduce 10.33.XX.2 para no permitir conexiones desde esa dirección.
- f. En *Admin Interface Settings* introduce la contraseña de acceso “aaaaaa”, y deja el puerto por defecto.
- g. En *Logging* habilita “*Enable logging to file*”, habilita “*Limit log file size to*”, e introduce 200, habilita “*Use a different logfile each day*” y “*Delete old logfiles after*” e introduce 10.
- h. En *Speed Limits* habilita “*Use speed limits rules*” en “*Download Speed Limit*” y añade una regla de 10KB/s de lunes a viernes.
- i. En *Speed Limits* habilita “*Use speed limits rules*” en “*Upload Speed Limit*” y añade una regla de 5KB/s de lunes a viernes y de 15KB/s el sábado y el domingo.
- j. En *Autoban* habilita “*Enable automatics ban*” e introduce 10 y 2 respectivamente para “*banear*” durante 2 horas a las conexiones desde direcciones IP que fallen 10 veces en la autenticación al conectarse al servidor.

1.6. Haz clic en OK para aplicar la configuración.

2. Configuración de usuarios y grupos

El servidor FTP *Filezilla* solo permite crear usuarios virtuales, es decir los usuarios con los que es posible establecer conexiones al servidor FTP son usuarios propios de servidor FTP y no del sistema operativo donde se ejecuta. Otro aspecto a tener en cuenta es que los usuarios son “enjaulados” en su directorio predeterminado cuando se conectan por FTP.

- 2.1. Crea los directorios **C:\ftp**, **C:\ftp\pub** y **C:\ftp\alumnos**. Crear varios ficheros dentro de los directorios.
- 2.2. En la barra de tareas accede a **Edit, Users** para configurar los usuarios virtuales y sus privilegios.

- 2.3. En la página **General** haz clic en **Add** y crea el usuario **anonymous** sin que pertenezca a un grupo (*none*).
- 2.4. Con el usuario **anonymous** seleccionado accede a la página **Shared Folders**, haz clic en **Add**, selecciona el directorio **C:\ftp\pub** y deja los permisos por defecto (*Files ⇒ Read y Directories ⇒ List, +Subdirs*).
- 2.5. Accede a la pagina **General** haz clic en **Add** y crea el usuario **profesor** sin que pertenezca a un grupo (*none*). Habilita la opción **Password** e introduce una contraseña.
- 2.6. Con el usuario **profesor** seleccionado accede a la página **Shared Folders**, haz clic en **Add**, selecciona el directorio **C:\ftp** y marca todos los permisos (*Files ⇒ Read, Write, Delete, Append y Directories ⇒ Create, Delete, List, +Subdirs*).
- 2.7. Haz clic en **OK** para aplicar los cambios.
- 2.8. En la barra de tareas accede a **Edit, Groups** para configurar los grupos virtuales y sus privilegios.
- 2.9. En la página **General** haz clic en **Add** y crea el grupo **alumnos**.
- 2.10. Con el grupo **alumnos** seleccionado accede a la página **Shared Folders**, haz clic en **Add**, selecciona el directorio **C:\ftp\alumnos** y establece los siguientes permisos (*Files ⇒ Read, Write y Directories ⇒ Create, List, +Subdirs*).
- 2.11. Con el grupo **alumnos** seleccionado accede a la página **Speed Limits** y establece como velocidad de subida 2KB/s.
- 2.12. Haz clic en **Ok** para aplicar los cambios.
- 2.13. En la barra de tareas accede a **Edit, Users** para configurar los usuarios virtuales y sus privilegios.
- 2.14. Accede a la pagina **General** haz clic en **Add** y crea el usuario **alumno1** de forma que pertenezca al grupo **alumnos**. Habilita la opción **Password** e introduce una contraseña.
- 2.15. Accede a la pagina **General** haz clic en **Add** y crea el usuario **alumno2** de forma que pertenezca al grupo **alumnos**. Habilita la opción **Password** e introduce una contraseña.
- 2.16. Haz clic en **OK** para aplicar los cambios.

3. Configuración del *Firewall* de Windows 2008

Hay que configurar el *Firewall* de **w2008XX** para que permita conexiones al servidor FTP. Utilizaremos la opción “*Dejar pasar un programa*” que configura adecuadamente los puertos de entrada y de salida.

- 3.1. Accede a la ventana de configuración básica del *Firewall* de Windows (Menú Inicio, Panel de control, Seguridad, Dejar pasar un programa a través de Firewall de Windows).
- 3.2. Haz clic en **Agregar un programa**, busca y selecciona **C:\Archivos de Programa\FileZilla Server\FileZilla server.exe** y acepta los cambios, Figura 4.10.

4. Comprobar la configuración

- 4.1. Inicia sesión en **ubuntuXX**.
- 4.2. Utiliza el cliente FTP *Filezilla* para conectarte al servidor instalado en **w2008XX** y prueba la configuración del servidor (prueba usando la IP del equipo y su nombre DNS). Conéctate con los usuarios **anonymous**, **profesor**, **alumno1** o **alumno2** y prueba a subir, descargar, borrar, etc., véase Figura 4.11

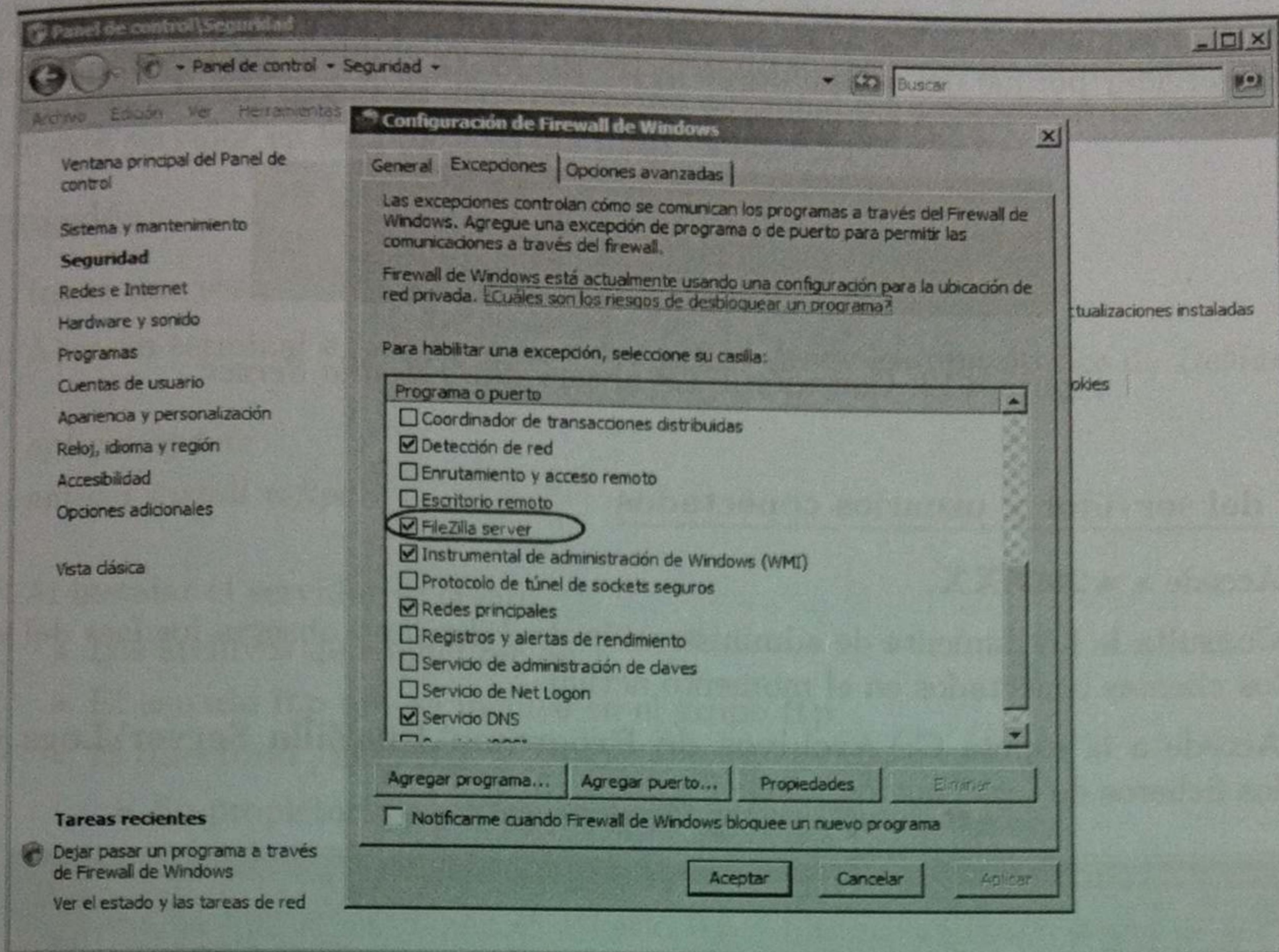


Figura 4.10: Configuración de *Filezilla Server* en el *Firewall de Windows 2008*

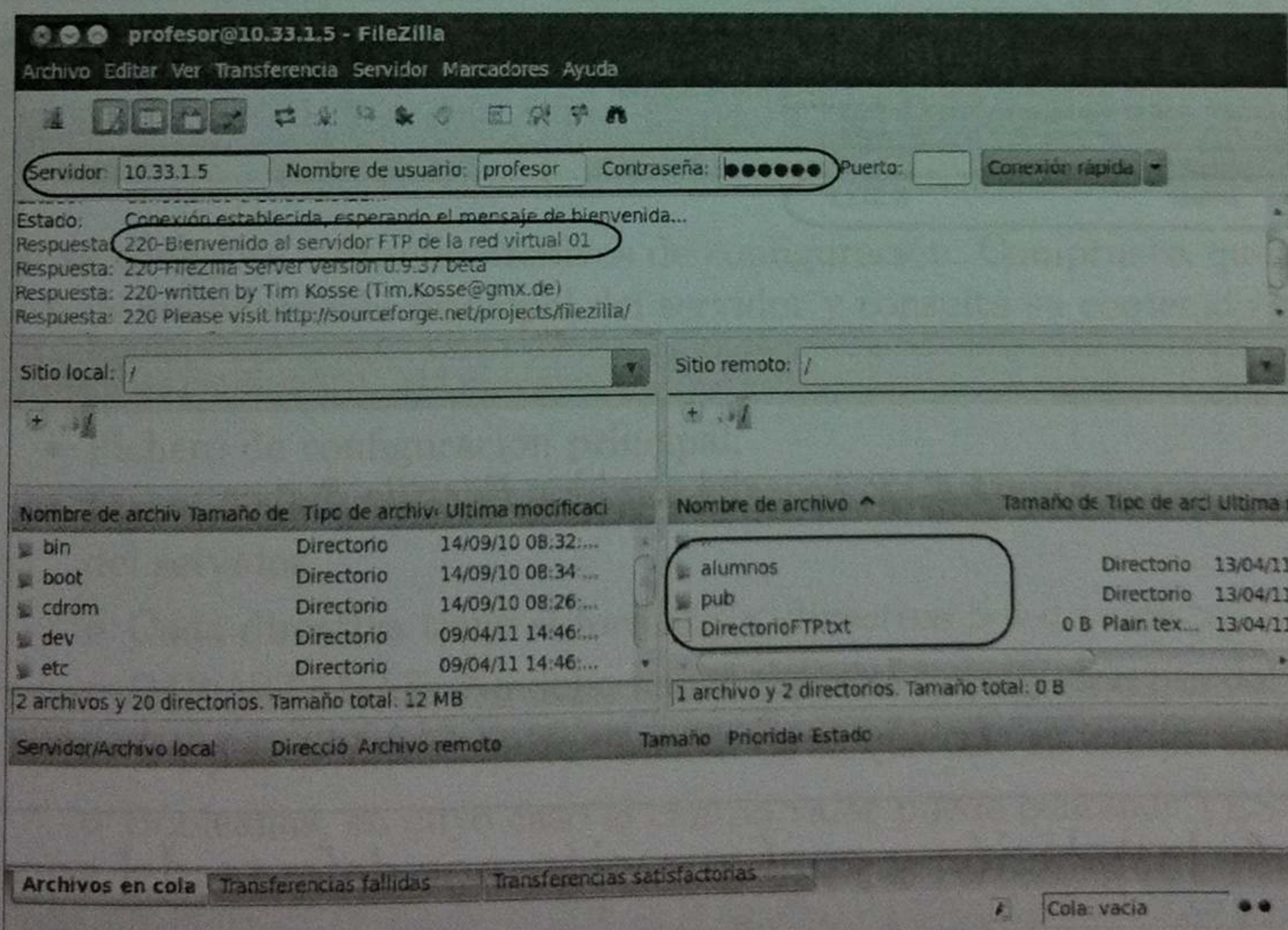


Figura 4.11: Conexión FTP al servidor *Filezilla Server*

4.3. Inicia sesión en **debianXX**.

4.4. Utiliza el cliente FTP en línea de comandos y comprueba que no es posible conectarse al servidor porque del acceso desde la IP 10.33.XX.2 no está permitido, véase Figura 4.12.

```
root@debian01:~# ftp 10.33.1.5
Connected to 10.33.1.5.
550 No connections allowed from your IP
ftp> -
```

Figura 4.12: Conexión FTP al servidor *Filezilla Server*

5. Logs del servidor y usuarios conectados

5.1. Accede a **w2008XX**.

5.2. Consulta la herramienta de administración de *Filezilla* y observa los *logs* del servidor y los clientes conectados en el momento actual.

5.3. Accede a la capeta **C:\Archivos de Programa\FileZilla Server\Logs** y observa los ficheros de *logs* creados, véase Figura 4.13.

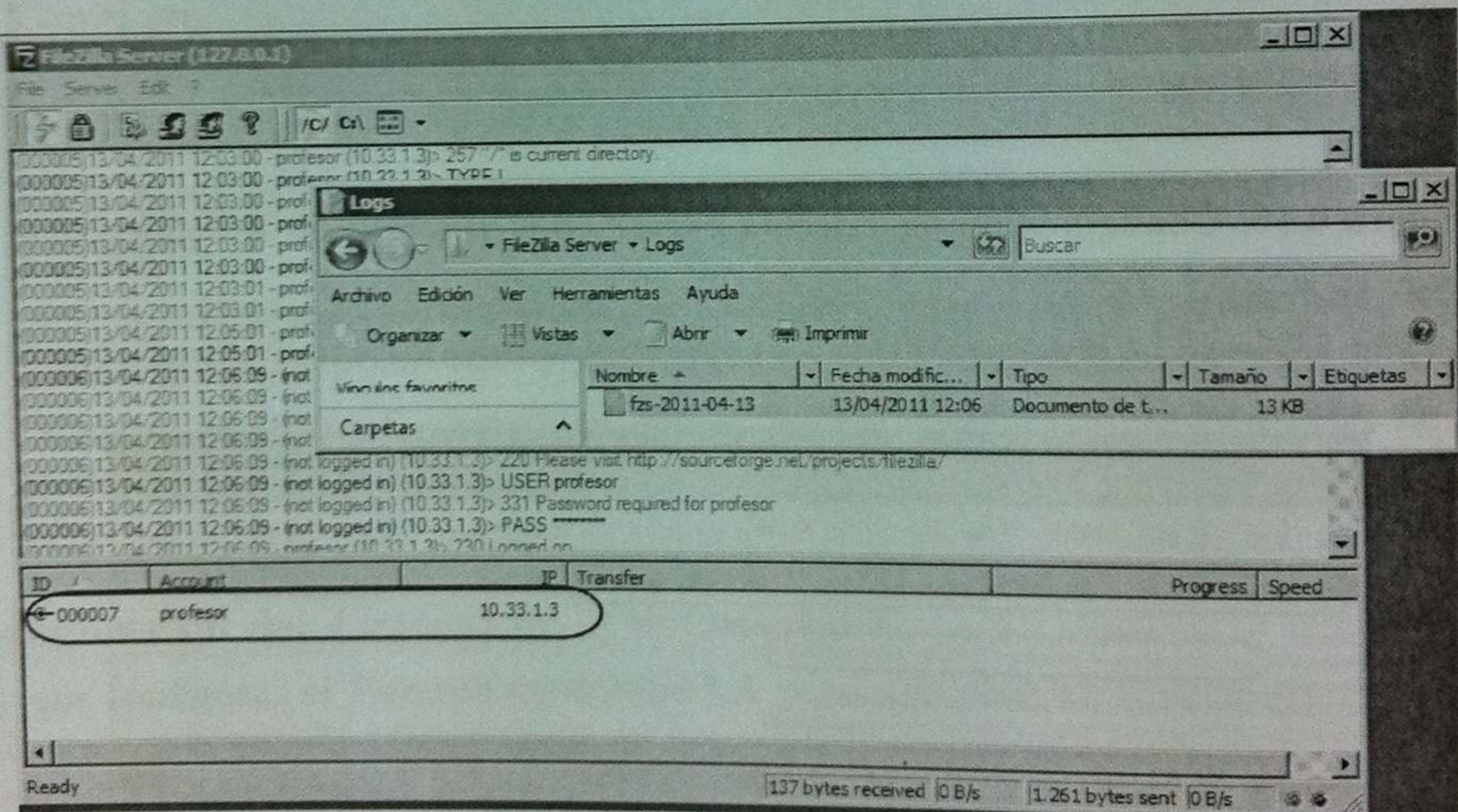


Figura 4.13: *Logs* del servidor *Filezilla Server*

Práctica 4.4: Instalación y configuración por defecto del servidor *vsftpd* en *Linux*

Instala, analiza y prueba la configuración por defecto del servidor *vsftpd* en **debianXX** (la configuración en **ubuntuXX** sería similar).

Very Secure FTP Daemon (vsftpd) (<http://vsftpd.beasts.org/>) es un servidor FTP rápido, seguro y fácil de configurar que se distribuye con licencia GNU (*General Public License*) para sistemas *Unix* y *Linux*. Está disponible en los repositorios de software de las principales distribuciones de *Linux*. Recomendamos consultar documentación adicional para completar las explicaciones que se hacen de este servidor en las prácticas del libro.

1. Instalación

- 1.1. Inicia una sesión en **debianXX** con el usuario **root**.
- 1.2. Abre un terminal e instala el servidor *vsftpd* desde los repositorios de *Debian*.

```
apt-get update
```

```
apt-get install vsftpd
```

Al instalar el servidor se crean:

- Los archivos de configuración.
- El usuario **ftp** que se incluye en el grupo **ftp**.
- El directorio **/srv/ftp**
 - Su propietario es el usuario **root** y su grupo es **ftp**.
 - El directorio predeterminado de los usuarios anónimos.

- 1.3. Comprueba que se ha creado el usuario **ftp** y que su directorio *home* es **/srv/ftp**.

```
cat /etc/passwd
```

```
cat /etc/group
```

- 1.4. Comprueba que se ha creado el directorio **/srv/ftp** y que su propietario es el usuario **root** y su grupo es **ftp**.

```
ls -l /srv
```

- 1.5. *vsftpd* se configura editando archivos de configuración. Comprueba que se ha creado el archivo de configuración principal del servidor y consulta su contenido:

- **/etc/vsftpd.conf**
 - Fichero de configuración principal.
 - El archivo contiene un conjunto de directivas que determinan el comportamiento del servidor.
 - Cada directiva tiene el formato *<directiva>=<valor>*
 - IMPORTANTE: No debe haber espacios antes y después del signo “=”.
 - Tres tipos de directivas en función de lo que vale su campo valor
 - Booleanas, en cuyo caso el campo valor puede contener YES o NO.
 - Numéricas.
 - De cadena.
 - Los comentarios son a nivel de línea y se utiliza el símbolo almohadilla (#).
 - Las directivas que no se especifiquen en el fichero de configuración, utilizan su valor por defecto.

- Consultar *man vsftpd.conf* para obtener información sobre toda las directivas.
- Otros ficheros de configuración:
 - Su nombre y ubicación se pueden definir en las directivas de */etc/vsftpd.conf*.
 - Algunos que usaremos en prácticas posteriores son
 - */etc/ftpusers*
 - */etc/vsftpd.user_list* (no está creado por defecto)
 - */etc/vsftpd.chroot_list* (no está creado por defecto)

1.6. Comprueba que el servidor está iniciado.

```
ps -ef | grep vsftpd
```

1.7. Comprueba que el servidor está escuchando en el puerto TCP 21.

```
netstat -ltn
```

1.8. Haz una copia de seguridad del fichero de configuración principal que se modificarán en esta y sucesivas prácticas (*/etc/vsftpd.conf*).

2. Usuarios y preparación para probar el servidor

vsftpd permite la conexión de:

▪ Usuarios anónimos.

- Si está habilitado (directiva *anonymous_enable*) el servidor *vsftpd* permite la conexión de usuarios anónimos con el nombre de usuario **anonymous** o **ftp**.
- Cuando se conecta un usuario anónimo al servidor entra en el directorio, por defecto */srv/ftp*, especificado en */etc/passwd* para el usuario **ftp**.
- Este directorio es, a efectos del usuario anónimo, su directorio raíz. Esta “enjaulado” en él.

▪ Usuarios locales con cuenta en el sistema (*/etc/passwd*).

- Si está habilitado (directiva *local_enable*) el servidor *vsftpd* permite la conexión de usuarios locales con cuenta en el sistema.
- Cuando se conecta un usuario local hay dos opciones de configuración.
 - Que no se le “enjaule” en su directorio **home**. Puede acceder al resto del árbol de directorio en función de los permisos definidos.
 - Que se le “enjaule” en su directorio **home**. No tiene disponible el resto del sistema de archivos.

▪ Usuarios virtuales.

- Es posible crear cuentas de usuarios virtuales (que no existan en el sistema operativo).
- Las cuentas de usuario, se almacenan en ficheros o en bases de datos, servicios de directorio, etc., pueden ser consultados por el servidor para realizar la autenticación.
- Las cuentas virtuales se mapean en un usuario local del sistema.

Vamos a crear usuarios locales en el sistema para probar posteriormente el funcionamiento del servidor.

2.1. Accede al directorio **/srv/ftp** y crea tres archivos de texto.

2.2. Crea los usuarios locales **mortadelo y **filemon**:**

*adduser mortadelo
adduser filemon*

2.3. Inicia sesión en **debianXX como usuario **mortadelo** y crea en su directorio *home* dos archivos de texto. Cierra la sesión.**

2.4. Inicia sesión en **debianXX como usuario **filemon** y crea en su directorio *home* dos archivos de texto. Cierra la sesión.**

3. Configuración por defecto

La configuración del servidor por defecto es:

- Permite solo el acceso a usuarios anónimos.
 - Nombre de usuario **anonymous** o **ftp**.
 - *Password*: en blanco.
- Los usuarios anónimos están “enjaulados” en **/srv/ftp**.
- Pueden descargar archivos (con permisos de lectura para “otros”).
- No pueden subir archivos.
- El servidor usa todo el ancho de banda disponible.
- El fichero de *logs* por defecto es **/var/log/vsftpd.log**.

3.1. Inicia sesión en **debianXX como usuario **root**.**

3.2. Consulta el fichero de configuración de servidor (/etc/vsftpd.conf**), Figura 4.14. Familiarízate con su formato y sus directivas y comprueba que:**

- Está habilitado el acceso a los usuario anónimos (directiva *anonymous_enable*).

anonymous_enable=YES

- Está deshabilitado el acceso a los usuarios locales (directiva *local_enable*).

local_enable=NO

- No se permite subir archivos al servidor (directiva *write_enable*).

write_enable=NO

3.3. Inicia sesión en **w7XX o **ubuntuXX** y conéctate al servidor ftp usando el cliente ftp que quieras. Verifica que:**

- a. Es posible acceder como usuario anónimo y descargar archivos. Observa que el usuario anónimo esta “enjaulado”.
- b. No es posible acceder con los usuarios **mortadelo** y **filemon** por ser usuarios locales.

```

# Run standalone? vsftpd can run either from an inetd or as a standalone
# daemon started from an initscript.
listen=YES

#
# Run standalone with IPv6?
# Like the listen parameter, except vsftpd will listen on an IPv6 socket
# instead of an IPv4 one. This parameter and the listen parameter are mutually
# exclusive.
#listen_ipv6=YES

#
# Allow anonymous FTP? (Beware - allowed by default if you comment this out).
anonymous_enable=YES

#
# Uncomment this to allow local users to log in.
#local_enable=YES

#
# Uncomment this to enable any form of FTP write command.
#write_enable=YES

#
# Default umask for local users is 077. You may wish to change this to 022.
#umask=077

```

Figura 4.14: Fichero /etc/vsftpd.conf

4. Logs del servidor

4.1. Accede a **debianXX**.

4.2. Consulta el fichero **/var/log/vsftpd.log** y observa los accesos al servidor y las transferencias realizadas.



Práctica 4.5: Configuración básica del servidor *vsftpd* en Linux

Configura el servidor *vsftpd* instalado en **debianXX** con las siguientes opciones:

- Se permitirá la conexión a los usuarios locales.
- Los usuarios locales podrán descargar y subir archivos.
- Los usuarios locales estarán “enjaulados” en su directorio *home*.

Conéctate al servidor FTP desde clientes FTP instalados en otros equipos de la red y comprueba que se cumple la configuración realizada.

Para realizar la configuración del servidor *vsftpd* es posible usar el fichero de configuración por defecto y comentar/descomentar y/o modificar/añadir nuevas directivas o borrar el fichero y configurar las directivas necesarias. En esta práctica vamos a optar por la primera opción.

1. Inicia sesión en **debianXX** como usuario **root**.

2. Modifica el fichero de configuración del servidor para que los usuarios locales puedan conectarse al servidor (directiva *local_enable*).

local_enable=YES

3. Reinicia el servidor para que se lea de nuevo el fichero de configuración y se apliquen los cambios.

```
/etc/init.d/vsftpd stop  
/etc/init.d/vsftpd start
```

4. Comprueba que el servidor está iniciado (si hay errores en el fichero de configuración el servidor no se iniciará).

```
ps -ef | grep vsftpd
```

5. Inicia sesión en **w7XX** o **ubuntuXX** y conéctate al servidor ftp usando el cliente FTP que quieras. Verifica que:

- 5.1. Es posible acceder como usuario anónimo y descargar archivos. Observa que el usuario **anónimo** esta “enjaulado”.
- 5.2. Es posible acceder con el usuario **mortadelo**. Comprueba que no está enjaulado en su home, puede descargar archivos pero no puede subir archivos.
6. Modifica el fichero de configuración del servidor para que los usuarios locales puedan subir archivos al servidor (directiva *write_enable*)

write_enable=YES

7. Reinicia el servidor para que se lea de nuevo el fichero de configuración y se apliquen los cambios. Comprueba que el servidor está iniciado.

8. Inicia sesión en **w7XX** o **ubuntuXX** y conéctate al servidor ftp usando el cliente ftp que quieras. Verifica que es posible acceder con el usuario **mortadelo** y que ahora puede subir archivos.
9. Modifica el fichero de configuración del servidor ftp para que los usuarios locales sean “enjaulados” en su directorio **home** (directiva *chroot_local_user*).

chroot_local_user=YES

10. Reinicia el servidor para que se lea de nuevo el fichero de configuración y se apliquen los cambios. Comprueba que el servidor está iniciado.
11. Inicia sesión en **w7XX** o **ubuntuXX** y conéctate al servidor ftp usando el cliente ftp que quieras. Verifica que es posible acceder con el usuario **mortadelo** y que está enjaulado en su directorio **home**.
12. Consulta el fichero de *log* del servidor **/var/log/vsftpd.log** y comprueba que se han registrado los accesos y las transferencias realizadas.

