

***Práctica 3:
Puesta en marcha de un
servidor FTP***

Arquitectura de Sistemas y Aplicaciones Distribuidas – U.L.P.G.C.

Índice

I. Introducción al protocolo FTP.....	2
Características Técnicas del Protocolo FTP.....	3
Funcionamiento Lógico del Protocolo FTP.....	4
Comandos del Protocolo FTP.....	5
II. Obtención e Instalación del servidor de ficheros Pure-FTPd.....	10
III. Configuración del superdemonio xinetd.....	11
IV. Configuración de distintas formas de acceder al servidor FTP.....	13
Acceso en modo usuario del sistema.....	14
Acceso como usuario anónimo.....	21
Acceso como usuarios virtuales.....	33
Aceptando usuarios virtuales.....	34
Creando un usuario.....	34
Cambiando opciones de los usuarios.....	35
Mostrando y borarando usuarios.....	35
Aplicando los cambios.....	36
V. Creación de un Servidor Virtual.....	37

Introducción al protocolo FTP

En esta práctica realizaremos la configuración de un servidor FTP, para permitir la Transferencia de Ficheros entre el servidor y los clientes FTP que se conectan al mismo. En el proceso de la transferencia se usará el protocolo FTP (File Transfer Protocol – Protocolo de Transferencia de Ficheros).

En 1971 se creó con un modelo de transferencia llamado RFC 141 en M.I.T. Fue hasta después de muchas revisiones que llegó a RFC 265 cuando ya se le considera un protocolo de transferencia de archivos completo entre HOSTs (servidores de archivos) de ARPHANET. Al final de la edición de RFC 765 se incluyeron algunos de los que son ahora los comandos de este protocolo.

FTP es uno de los servicios más útiles a la hora de transmitir y recibir ficheros de cualquier tipo. Funciona con protocolo TCP/IP, que permite acceder al servidor para recibir o transmitir ficheros. El protocolo FTP permite la conexión entre máquinas clientes y un servidor para la descarga de ficheros, por lo general, tras un proceso de autenticación o **login**, introduciendo el nombre de usuario y la contraseña (password). También es posible acceder sin necesidad de **logearse**, es decir, accediendo a un FTP **anónimo**.

Un servidor FTP es un gran contenedor, en el cual podemos encontrar gran cantidad de archivos y directorios para diferentes Sistemas Operativos. Los objetivos principales de este protocolo son:

1. Posibilitar la compartición de archivos entre computadoras (programas y/o datos)
2. Posibilitar el uso remoto de las computadoras
3. Transferir datos de una forma segura y optima entre computadoras.

El protocolo FTP, más que para ser usado por un usuario directamente es usado por los programas para comunicarse, lo que facilita al usuario despreocuparse de las características del sistema con que conecta. Algunas de las ventajas de usar FTP son:

1. **Interactivo:** el cliente FTP permite al usuario conectado navegar por los directorios del servidor y ejecutar comandos que harán más fácil la transferencia. Según el cliente FTP las prestaciones podrán ser mayores o no.

2. **Formato:** permite la transferencia de varios tipos de datos.
3. **Autenticación FTP:** todos los usuarios de Internet pueden acceder a todos los lugares

públicos, con solo su nombre de usuario.

4. **FTP Anónimo:** algunos servidores permiten conectarse como usuario anónimo.
5. **Velocidad:** es confiable y rápido para enviar y recibir archivos.

Características Técnicas del Protocolo FTP

Los tipos de datos en la transferencia por FTP son:

1. El tipo **ASCII**, es el mas común en el protocolo FTP. Se usa cuando se transfieren archivos de texto, la computadora que envía (**sender**), debe convertir cualquiera que sea su estructura de archivos interna, debe convertir sus datos al formato genérico de 8 bits, y el que recibe (**receiver**) lo debe convertir de nuevo a su formato propio.

2. El tipo **EBCDIC** es el mas eficiente cuando ambos el que recibe y el que envía lo usan como formato propio, este tipo se representa también en 8 bits pero de forma **EBCDIC**. Lo único en lo que cambian es en la forma de reconocer los códigos de los caracteres.

3. El formato de **IMAGEN** es cuando se compacta todo lo que se quiere enviar en cadenas seguidas de paquetes de 8 bits, esto es no importa el formato en que internamente se maneje la información, cuando se va a enviar se tiene que hacer una conversión de 8 bits en 8 bits y cuando el que recibe tiene todo el paquete, el mismo debe codificarlos de nuevo para que la transmisión sea completada.

4. El formato **LOCAL**, en el que los datos se transfieren en bytes lógicos del tamaño especificado por el segundo parámetro obligatorio, tamaño de byte, que debe ser un entero decimal.

En la estructura de datos en FTP se consideran tres tipos diferentes de archivos:

1. **File:** estructura donde no hay estructuras internas y el archivo es considerado una secuencia continua de bytes.

2. **Record:** estructura la cual debe soportar todas las implementaciones del FTP para ficheros de texto (tipos ASCII o EBCDIC). El fichero está formado por registros dispuestos secuencialmente.

3. **Page:** estructura donde los archivos contienen paginas enteras indexadas separadas.

En FTP nos encontramos con tres modos de transferencia:

1. **Modo flujo (stream):** en el que los datos se transmiten como un flujo de bytes. En un fichero estructurado en registros el primer byte del código de control consistirá en todo unos, el

carácter de escape. El segundo byte valdrá 1 para EOR y 2 para EOF y 3 para indicar ambos. Si la estructura es de fichero, se indica el EOF cuando el ordenador que envía los datos cierra la conexión de datos.

2. **Modo bloque (block):** en el que el fichero se transmite como una serie de bloques de datos precedidos por uno o más bytes cabecera. Los bytes de la cabecera contienen un campo contador y un código descriptor. El campo contador indica la longitud total del bloque de datos en bytes y el código descriptor define: último bloque del fichero (EOF), último bloque del registro (EOR), indicador de reinicio o datos sospechosos.

3. **Modo comprimido:** muy útil para obtener un ancho de banda adicional en transmisiones muy largas. Hay tres clases de información a enviar: datos normales, enviados en una cadena de bytes; datos comprimidos, formados por repeticiones o relleno; e información de control, enviada en una secuencia de escape de dos bytes.

Funcionamiento Lógico del Protocolo FTP

El intérprete de protocolo del servidor debe "escuchar" en el Puerto destinado a la conexión de control y que por defecto es el 21. El usuario o el intérprete de protocolo de usuario iniciará la conexión de control full-duplex (bidireccional). Los procesos de servidor y de usuario deberían seguir las convenciones del Protocolo Telnet. El servidor deberá cerrar la conexión a petición del usuario una vez que todas las transferencias y respuestas se han enviado.

Una vez establecida la conexión de control y antes de que tenga lugar la de datos es preciso efectuar el proceso de autenticación del cliente, que enviará los comandos USER y PASS por la conexión de control con el fin de identificarse con el servidor.

La mecánica de transferir datos consiste en preparar la conexión de datos en los puertos apropiados y elegir los parámetros para la transferencia. El usuario y los server-DTPs tienen ambos un puerto por defecto. El puerto por defecto del proceso de usuario es el mismo que el puerto de la conexión de control. El puerto por defecto del proceso servidor es el puerto adyacente al puerto de la conexión de control, es decir, el puerto 20.

El tamaño de byte para la transferencia es de 8 bits. Este tamaño sólo es relevante para la transferencia de datos.

Todas las implementaciones del FTP deben soportar el uso de los puertos de datos por defecto y sólo el user-PI puede solicitar un cambio a un puerto diferente. Para ello usa la orden PORT, que permite al usuario especificar un puerto alternativo. Además, el user-PI puede solicitar

al servidor la localización de un puerto en el propio servidor con la orden PASV.

En general, es responsabilidad del servidor mantener la conexión de datos (abrirla y cerrarla).

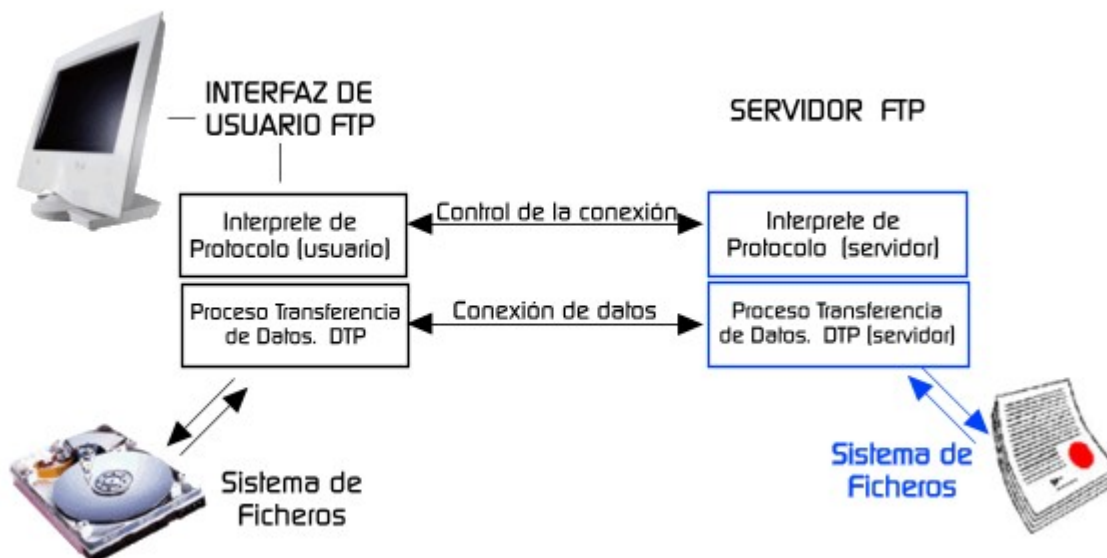


Illustration 1: Arquitectura del Protocolo FTP

Para abortar una transferencia de archivo, se utiliza la clave de interrupción de terminal (generalmente Ctrl-C). Las transferencias que se estén realizando se paran inmediatamente. El protocolo ftp para el comienzo (o recepción) de transferencias enviando primero un comando ABORT al servidor remoto y rechaza cualquier información posterior recibida. La velocidad con que esto sea logrado depende de la capacidad del servidor remoto para procesar el comando ABORT. Si el servidor remoto no tolera ABORT, el cursor (**ftp>**) no aparece hasta que el servidor remoto complete el envío del archivo solicitado.

Una de las diferencias principales que existe entre ftp y Telnet está en que mientras con Telnet podemos ver los ficheros en la pantalla de nuestro ordenador, con ftp los tenemos que traer desde el lugar al que estamos conectados hasta nuestro ordenador y después verlos.

Comandos del Protocolo FTP

La mayoría de los servidores de ficheros trabajan en UNIX. Por ello, los ficheros y los directorios tienen nombres con la convención UNIX pero al transferirlos a un PC adoptan automáticamente el formato del sistema operativo que posea el mismo. El nombre del servidor que comunica con ftp puede ser especificado en la línea de comando. Si se especifica el servidor, ftp abre inmediatamente una conexión con el mismo (véase el comando "open"), de lo contrario, ftp

espera las instrucciones del usuario.

El protocolo de transferencia de archivos especifica parámetros de transferencia de archivo de tipo, modo, forma y estructura. El protocolo ftp soporta los tipos de archivos ASCII y binario. ASCII es el tipo ftp por defecto. Los comandos más comunes que podemos utilizar son los siguientes :

user argumento --> El argumento es una cadena Telnet que identifica al usuario. Normalmente esta será la primera orden a transmitir una vez establecida la conexión de control.

pass argumento --> El argumento es una cadena Telnet especificando la contraseña del usuario. Esta orden debe ir inmediatamente después de la orden user.

acct argumento --> Se usa en ocasiones, cuando se puede requerir una cuenta para acceder (PASS) y otra sólo para cierto tipo de acceso, como almacenar ficheros. En este último caso, la orden se puede enviar en cualquier momento.

port argumentos --> Esta orden permite al usuario cambiar el puerto, aunque normalmente hay puertos por defecto, lo que hace que esta orden no sea imprescindible. Si se usa esta orden, el argumento es la concatenación de una dirección IP (32 bits) y un puerto TCP (16 bits).

pasv respuesta --> Esta orden solicita al server-DTP que "escuche" en un puerto de datos (que no es el puerto por defecto) y espere a recibir una conexión en lugar de iniciar una al recibir una orden de transferencia. La respuesta a este comando incluye la dirección IP y el puerto donde este servidor está esperando a recibir la conexión.

type [nombre del tipo] --> Establece el tipo de transferencia de archivo de ftp para el nombre del tipo. El nombre de tipo puede ser A - ASCII, B - BCDIC, I - image o L - local.

stru argumento --> El argumento es un único carácter Telnet especificando una estructura de fichero. Este puede ser: F - Fichero (sin estructurar en registros) R - Estructurado en registros P - Estructurado en páginas. La estructura por defecto es Fichero.

mode argumento --> El argumento es un único carácter Telnet especificando un modo de transferencia. Los posibles códigos son los siguientes: S - Flujo, B - Bloque, C - Comprimido. El modo por defecto es Flujo.

append archivo local [archivo remoto] --> Copia el archivo local al final del archivo remoto.

retr ruta --> Transfiere una copia del fichero especificado en el nombre de ruta al proceso que está al otro lado de la conexión de datos.

stor ruta --> Permite leer los datos transferidos por la conexión de datos y los guarde en un

fichero, especificado en la ruta, en el servidor, si este fichero existe sustituye su contenido, si no existe lo crea.

stou --> Esta orden se comporta igual que STOR sólo que el fichero resultante se crea en el directorio actual con un nombre único para ese directorio.

ascii --> Transfiere los archivos en modo ASCII. Este es el valor por defecto.

binary --> Transfiere los archivos en modo binario.

bye --> Cierra la conexión con el huésped servidor. También, tecleando los caracteres de final de archivo (EOF) finaliza la sesión.

cd directorio remoto --> Se introduce dentro del "directorio remoto" del servidor.

chmod modo-nombre del archivo --> Cambia los permisos del archivo "nombre del archivo" en el sistema remoto al modo indicado con la instrucción.

close --> Termina la conexión con el servidor. El comando close no sale de ftp.

cr Cambia el retorno de carro alterado durante la recuperación de un archivo ascii.

delete archivo remoto --> Elimina el archivo remoto. El archivo remoto puede ser un directorio vacío.

dir [directorio remoto] [archivo local] --> Escribe un listado del directorio remoto u opcionalmente de un archivo local. Si ni el archivo local ni el directorio remoto se especifican, lista el directorio de trabajo actual.

disconnect --> Un sinónimo de close.

get archivo remoto [archivo local] --> Copia el archivo remoto al archivo local. Si el archivo local no se especifica, ftp utiliza el nombre del archivo remoto especificado como el nombre de archivo local.

hash --> Conmuta imprimiendo un signo de fragmentación (#) cada 1024 bytes transferidos.

help [comando] --> Imprime un mensaje informativo del comando de ftp llamado. Si no se especifica el comando, nos da una lista de todos los comandos de ftp.

lcd [directorio local] --> Sitúa el directorio de trabajo en el directorio local señalado. Si el directorio local no se especifica, se sitúa en el directorio de trabajo local del usuario.

ls [directorio remoto] [archivo local] --> Escribe un listado del directorio remoto en el archivo local. El listado incluye toda la información del sistema dependiente que el servidor quiera incluir; por ejemplo, la que la mayoría de los sistemas de UNIX producen con el comando ls -l (ver también nlist). Si ni el archivo local ni el directorio remoto se especifican, lista el directorio de trabajo remoto.

mdelete [archivos remotos] --> Elimina los archivos remotos.

mdir archivos remotos archivo local --> Escribe un listado de los archivos remotos en el archivo local.

mget archivos remotos --> Copia los archivos remotos en el sistema local.

mkdir nombre de directorio --> Crea el nombre del directorio remoto.

mls archivos remotos archivo local --> Escribe un listado abreviado de archivos remotos en el archivo local.

modtime archivo remoto --> Muestra la fecha de la última modificación del archivo remoto.

mput archivo local --> Copia el archivo local del sistema local al sistema remoto.

newer nombre de archivo --> Elige el archivo sólo si la fecha de modificación del archivo remoto es más reciente que el archivo del sistema actual. Si el archivo no existe en el sistema actual, el archivo remoto es considerado más reciente. Por lo demás, este comando es idéntico a "get".

nlist [directorio remoto] [archivo local] --> Escribe un listado abreviado del directorio remoto en el archivo local. Si el directorio remoto no ha quedado especificado, se utiliza el directorio de trabajo actual.

open servidor-huésped [número de puerto] --> Establece una conexión entre servidor-huésped, utilizando el número del puerto (si se especifica). Si el auto-login está permitido, ftp intenta entrar en el servidor.

put archivo local [archivo remoto] --> Copia el archivo local en el archivo remoto.

pwd --> Nos informa del nombre del directorio de trabajo actual.

quit --> Un sinónimo de bye.

quote argumentos --> Envía argumentos, al pie de la letra, al servidor.

recv archivo remoto [archivo local] --> Un sinónimo de get.

reget archivo remoto [archivo local] --> Reget suele actuar como get, excepto que si existe un archivo local y es más pequeño que el archivo remoto, el archivo local es supuesto como copia parcialmente transferida del archivo remoto y la transferencia continua desde el punto aparente de fallo. Este comando es útil cuando se transfieren archivos muy grandes en redes que tienden a interrumpir conexiones.

rhel [nombre del comando] --> Ayuda a solicitud del servidor. Si el nombre del comando es especificado, lo suministra al servidor.

rstatus [nombre de archivo] --> Sin argumentos, muestra el estado de la máquina remota.

Si se especifica el nombre de archivo, muestra el estado del nombre de archivo en la máquina remota.

rmdir directorio remoto --> Elimina el directorio remoto. El directorio remoto tiene que estar vacío.

rnfr argumento --> Esta orden indica el fichero que queremos cambiar de nombre en el servidor. Debe ir inmediatamente seguida de la orden "rnto" con el nuevo nombre para el fichero.

rnto argumento --> Esta orden especifica el nuevo nombre para el fichero indicado mediante el comando rnfr

send archivo local [archivo remoto] --> Un sinónimo de put.

size archivo remoto --> Muestra el tamaño del archivo remoto.

rerest argumento --> El argumento representa un marcador del servidor a partir del cual debe recomenzar la transferencia.

status --> Muestra el estado actual de ftp.

system --> Muestra el tipo de sistema operativo que posee la máquina remota.

? [comando] --> Un sinónimo de help. Imprime la información de ayuda del comando especificado.

Obtención e Instalación del servidor de ficheros Pure-FTPd

Para obtener el servidor de ficheros Pure-FTPd tendremos que ir a la dirección **www.pureftpd.org**. Desde esta página web, en la sección de **Downloads** conseguimos acceder a su FTP de descargas y finalmente descargamos la versión **1.0.20** de **Pure-FTPd**, desde el siguiente enlace:

<ftp://ftp.pureftpd.org/pub/pure-ftpd/releases/binary/pure-ftpd-1.0.19-1.i686.rpm>

El siguiente paso es la instalación de **pure-ftpd-1.0.19-1.i686.rpm**, para lo que basta usar el comando:

```
rpm -ihv pure-ftpd-1.0.19-1.i686.rpm
```

Configuración del superdemonio xinetd

En el equipo Pasarela procederemos a la configuración del superdemonio **xinetd**, con lo que se debe conseguir el funcionamiento básico del servidor FTP. Al realizar dicha configuración deberemos cumplir los dos siguientes requisitos:

1. Arranque del demonio bajo demanda.
2. Posibilidad de acceso remoto únicamente.

La configuración la haremos (en el caso de sistemas tipo Red Hat, como es la Fedora) en el fichero **/etc/xinetd.d/pure-ftpd**, que debemos de crear. La configuración que usaremos será la siguiente:

```
service ftp
{
    socket_type = stream
    server = /usr/local/sbin/pure-ftpd
    protocol = tcp
    user = root
    wait = no
    disable = no
}
```

Si usamos la opción **server_args = ARGUMENTOS** podremos pasar argumentos al superdemonio xinetd, de forma que el servidor FTP podrá ser configurado de diferentes formas. Esta opción la usaremos más adelante y variará en función de las configuraciones que apliquemos para el servidor FTP, que en realidad se hacen por comandos, y no mediante ficheros de configuración. No obstante se hará mención en ocasiones a las configuraciones estáticas, mediante ficheros. En concreto se hace uso del fichero **/etc/pure-ftpd.conf**. Pero como el servidor FTP no hace uso de dicho fichero, las configuraciones se harán por comandos de la forma antes explicada.

Para que se active el servicio FTP en la máquina tendremos que reiniciar el servidor, lo cual puede hacerse con el siguiente comando:

```
killall -USR2 xinetd
```

O de forma más correcta, reiniciando el servicio del superdemonio, es decir, con el comando siguiente:

```
service xinetd restart
```

Configuración de distintas formas de acceder al servidor FTP

Las distintas configuraciones se harán en el fichero `/etc/pure-ftpd.conf`, si se hace mención a configuraciones estáticas, pero por lo general las configuraciones se harán mediante el uso de comandos.

La lista de comandos estándar, que admite Pure-FTP es la siguiente, y será posteriormente usada para conseguir configurar correctamente el servidor FTP.

```
-0 --nottruncate
-1 --logpid <file>
-4 --ipv4only
-a --trustedgid <gid>
-A --chrooteveryone
-b --brokenclientscompatibility
-B --daemonize
-c --maxclientsnumber <number>
-C --maxclientsperip <number>
-d --verboselog
-D --displaydotfiles
-e --anonymousonly
-E --noanonymous
-f --syslogfacility <facility>
-F --fortunesfile <file>
-g --pidfile <path to pid file>
-G --norename
-h --help
-H --dontresolve
-i --anonymouscantupload
-I --maxidletime <time (min)>
-j --createhomedir
-k --maxdiskusagepct <percentage>
-K --keepallfiles
-l --login <auth> or <auth>:<config file>
-L --limitrecursion <number:number>
```

```
-m --maxload <load>
-M --anonymouscancreatedirs
-N --natmode
-o --uploadscript
-O --altlog <format>:<log file>
-p --passiveportrange <minport:maxport>
-P --forcepassiveip <ip address>
-q --anonymousratio <upload ratio>:<download ratio>
-Q --userratio <upload ratio>:<download ratio>
-r --autorename
-R --nochmod
-s --antiwarez
-S --bind <ip address,port>
-t --anonymousbandwidth <bandwidth (KB/s)>
-T --userbandwidth <bandwidth (KB/s)> or [<up bw>]: [<down bw>]
-u --minuid <uid>
-U --umask <mask>
-V --trustedip <ip address>
-w --allowuserfxp
-W --allowanonymousfxp
-x --prohibitdotfileswrite
-X --prohibitdotfilesread
-y --peruserlimits <per user max>:<max anonymous sessions>
-Y --tls <0:no TLS | 1:TLS+cleartext | 2:enforce TLS>
-z --allowdotfiles
-Z --customerproof
```

Acceso en modo usuario del sistema

Para que el acceso sea en modo usuario del sistema, se trata de conseguir que sólo entren al FTP los usuarios del sistema. Esto se consigue con la opción **-E**, que sólo acepta usuarios declarados en el sistema, es decir, no se deja que entren usuarios anónimos.

Una vez hecho esto se observa que los usuarios anónimos no pueden entrar al FTP:

```
[root@pasarela12 /]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
220-You are user number 1 of 50 allowed.
220-Local time is now 15:24. Server port: 21.
```

```
220-This is a private system - No anonymous login
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 60 seconds of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): anonymous
530 This is a private system - No anonymous login
Login failed.
421 Service not available, remote server has closed connection
```

Por su parte, los usuarios definidos en el sistema sí acceden sin problema. En el siguiente se accede como el usuario **enrique**, que se ha definido en el sistema.

```
[root@pasarela12 /]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
220-You are user number 1 of 50 allowed.
220-Local time is now 15:29. Server port: 21.
220-This is a private system - No anonymous login
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 60 seconds of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): enrique
331 User enrique OK. Password required
Password:
230-Your bandwidth usage is restricted
230-User enrique has group access to: enrique
230-OK. Current directory is /home/enrique
230 bienvenido! enrique.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Por otro lado, hay que destacar que el superusuario **root** siempre tiene acceso al FTP, bajo cualquier configuración operativa del FTP:

```
[root@pasarela12 /]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
220-You are user number 1 of 50 allowed.
220-Local time is now 15:30. Server port: 21.
220-This is a private system - No anonymous login
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 60 seconds of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): root
331 User root OK. Password required
Password:
230-Your bandwidth usage is restricted
230-User root has group access to: wheel disk adm sys daemon
230- bin root
230-OK. Current directory is /root
230 Bienvenido, señor don ROOT.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

Adicionalmente se deben cumplir las siguientes condiciones:

1. Permitir el acceso a los ficheros definidos por el sistema.

Se trata de que la opción **-A** o la opción **-a <GID>**. Con la opción **-a <GID>** se indica el **GID** indica el grupo de usuario que tiene acceso ilimitado por todo el árbol del sistema de ficheros, es decir, que puede salir fuera de su directorio personal. De este modo se limita a los miembros de un grupo la posibilidad de salir de su directorio, mientras que el resto no puede salir. Se conoce como **chroot** al hecho por el cual un usuario puede acceder a su directorio personal pero no puede acceder o salir fuera del mismo. No obstante, el usuario **root** (con uid 0) siempre tiene acceso total a todo el sistema de ficheros (incluso aunque esté fuera del grupo gid que se indique con la opción **-a <GID>**). Por otro lado, con la opción **-A** se consigue que ningún usuario pueda salir de su directorio personal de modo que no tendrá acceso a todos los ficheros definidos por el sistema.

En conclusión, para permitir el acceso a todos los ficheros definidos por el sistema hay que quitar las opciones **-a <GID>** y **-A**.

2. Desconexión en caso de inactividad.

Por comandos se hace uso de la opción **-I <MINUTOS>**. Así si en la opción `server_args` ponemos **-I 1** se echará al cliente transcurrido 1 minuto de inactividad.

Se usa la opción **MaxIdleTime <MINUTOS>** en fichero estático.

Una vez configurada esta opción con el valor **-I 1**, se observa que pasado un minuto, si intentamos ejecutar algún comando FTP, el servidor FTP lo rechaza indicando que hemos sido desconectados, pues habrá transcurrido más de 1 minuto de inactividad.

```
[root@pasarela12 /]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
220-You are user number 1 of 50 allowed.
220-Local time is now 15:32. Server port: 21.
220-This is a private system - No anonymous login
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 60 seconds of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): enrique
331 User enrique OK. Password required
Password:
230-Your bandwidth usage is restricted
230-User enrique has group access to: ususiste
230-OK. Current directory is /home/ enrique
230 bienvenido! enrique.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
421 Timeout - try typing a little faster next time
Passive mode refused. Turning off passive mode.
No control connection for command: Transport endpoint is not connected
ftp>
```

3. Los usuarios puedan salir de los directorios de los cuales son propietarios

Para hacerlo por comandos, si no deseamos que puedan salir de su directorio por defecto, deberemos usar la opción **-A**.

Se usa la opción **ChrootEveryone no** de forma estática.

4. Presentación de un mensaje de bienvenida cuando un usuario acceda al ftp.

Para que aparezca un mensaje de bienvenida tras logearse en el ftp, deberemos poner un fichero con el nombre **.banner** en la carpeta raíz o carpeta inicial que se abre cuando el usuario se conecta al ftp. De este modo se mostrará su contenido.

El directorio de inicio variará en función del usuario que entre al servidor FTP y de forma general depende del que se haya asignado al usuario. Los caso principales son:

- a) Superusuario o root. Su carpeta es **/root**, a no ser que se la cambie.
- b) Usuario del sistema. Por lo general, salvo que no se cambie, usará la carpeta **/home/USUARIO**, es decir, **/home** y una carpeta con su nombre.
- c) Usuario anónimo (anonymous). Por lo general, el sistema reserva la carpeta **/var/ftp**, para los usuarios anónimos.

De este modo, si el fichero **.banner** se encuentra en el directorio personal de ftp del usuario, en el directorio del usuario **root**, en directorio del usuario anónimo o en un servidor virtual, se imprimirá su contenido como mensaje de bienvenida.

El tamaño del fichero no podrá superar los 4000 bytes; en caso contrario no se mostrará.

Adicionalmente, en cada directorio puede ponerse un fichero **.message** de forma que cada vez que se accede al directorio que lo contiene se mostrará automáticamente un mensaje, que no es más que su contenido. Este mensaje puede ponerse incluso en el directorio inicial del ftp, pero al hacer el login no se mostrará, si bien, cuando nos hayamos movido por el árbol de directorios posteriormente, si accedemos de nuevo a dicho directorio, sí se mostrará el mensaje.

Finalmente, también es posible mostrar una cookie. Esto permite el muestreo de mensajes de bienvenida aleatorios. Las cookies son extraídas de un fichero de texto con el formato estándar de **fortune**. Esto obliga a instalar el paquete **fortune**, que estará en el directorio **/usr/share/fortune** habitualmente (no obstante, es posible usar cualquier fichero de texto que deseemos), y contendrá ficheros binarios (***.dat**) y de texto (sin la extensión **.dat**). De este modo, para usar las cookies de Pure-FTPd basta añadir el nombre de un fichero de texto a la opción **-F**.

Como ejemplo, a parte del resto de capturas, en el siguiente fragmento puede verse el mensaje de bienvenida para el usuario **enrique**, seguida del resto de mensajes antes de entrar en el intérprete de FTP.

```
230 bienvenido! enrique.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp> ls
```

5. Limitar el ancho de banda.

Mediante comandos se usa la opción siguiente: **-T <KBYTES/SEGUNDO> : <KBYTES/SEGUNDO>** para el ancho de banda de subida y de bajada, respectivamente.

En modo estático, se usa la opción **UserBandwidth <Kbytes/Segundo>**, que limitará el ancho de banda para todos los usuarios incluyendo al usuario anónimo. Esta velocidad incluirá a la velocidad de subida y de bajada conjuntamente.

Para hacer una prueba de esta opción, haremos uso de un fichero de cierto tamaño, que puede verse a continuación:

```
[root@pasarela12 ~]# ls -l ejemplo.pdf; ftp 172.16.1.6  
-rw-r--r-- 1 root root 463666 abr 18 15:40 ejemplo.pdf
```

Se trata del fichero **ejemplo.pdf** de **463666** bytes de tamaño, que es suficiente para hacer una prueba con la opción **-T 10:100**, que indica que:

Velocidad máxima de subida = 10 KB/s; Velocidad máxima de bajada = 100 KB/s

Obtendremos el siguiente resultado, en el que se sube el fichero y luego se descarga.

```
Connected to 172.16.1.6.  
220----- Welcome to Pure-FTPd [privsep] -----  
220-You are user number 1 of 50 allowed.  
220-Local time is now 15:44. Server port: 21.  
220-This is a private system - No anonymous login  
220-IPv6 connections are also welcome on this server.  
220 You will be disconnected after 60 seconds of inactivity.  
500 This security scheme is not implemented  
500 This security scheme is not implemented  
KERBEROS_V4 rejected as an authentication type  
Name (172.16.1.6:root): enrique  
331 User enrique OK. Password required  
Password:  
230-Your bandwidth usage is restricted
```

```
230-User enrique has group access to: ususiste
230-OK. Current directory is /home/ enrique
230 bienvenido! enrique.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
227 Entering Passive Mode (172,16,1,6,94,144)
150 Accepted data connection
226-Options: -l
226 0 matches total
ftp> put ejemplo.pdf
local: ejemplo.pdf remote: ejemplo.pdf
227 Entering Passive Mode (172,16,1,6,122,214)
150 Accepted data connection
226-File successfully transferred
226 45.282 seconds (measured here), 10.00 Kbytes per second
463666 bytes sent in 32 seconds (14 Kbytes/s)
ftp> ls
227 Entering Passive Mode (172,16,1,6,230,60)
150 Accepted data connection
-rw-r--r-- 1 enrique enrique 463666 Apr 18 15:45 ejemplo.pdf
226-Options: -l
226 1 matches total
ftp> get ejemplo.pdf
local: ejemplo.pdf remote: ejemplo.pdf
227 Entering Passive Mode (172,16,1,6,179,19)
150-Accepted data connection
150 452.8 kbytes to download
226-File successfully transferred
226 4.529 seconds (measured here), 99.98 Kbytes per second
463666 bytes received in 4.5 seconds (1e+02 Kbytes/s)
ftp>
```

Se puede ver que se indican dos tipos de velocidad, donde la velocidad del servidor FTP es la indicada como **measured here**, mientras que la otra es la del cliente, que sería la velocidad de descarga o subida que éste ve, y que depende de otros factores relevantes al tráfico de red de sus interfaces e, incluso, la carga de trabajo de su sistema.

Al subir el fichero se ve que se cumple que la velocidad de subida no supera los 10Kb/seg.

```
ftp> put ejemplo.pdf
local: ejemplo.pdf remote: ejemplo.pdf
227 Entering Passive Mode (172,16,1,6,122,214)
150 Accepted data connection
226-File successfully transferred
226 45.282 seconds (measured here), 10.00 Kbytes per second
463666 bytes sent in 32 seconds (14 Kbytes/s)
```

Al bajar el fichero se ve que se cumple que la velocidad de bajada no supera los 100Kb/seg.

```
ftp> get ejemplo.pdf
local: ejemplo.pdf remote: ejemplo.pdf
227 Entering Passive Mode (172,16,1,6,179,19)
150-Accepted data connection
150 452.8 kbytes to download
226-File successfully transferred
226 4.529 seconds (measured here), 99.98 Kbytes per second
463666 bytes received in 4.5 seconds (1e+02 Kbytes/s)
```

Las posibles opciones que se terminen por usar finalmente serían:

```
server_args = -E -I 1 -T 10:100
```

Dichas opciones irán en el campo **server_args** dentro del fichero de configuración del superdemonio **xinetd**.

Acceso como usuario anónimo

En primer lugar hay que verificar que se permita el acceso a usuarios anónimos, pues con el comando **-E** se prohíbe el acceso como usuario anónimo. Por tanto, debemos eliminar dicha opción si existiera. Además, para que el acceso sea sólo como usuario anónimo, deberemos hacer uso de la opción **-e**, de forma que cualquier usuario (**root** o usuarios del sistema, como **enrique**), aunque se indique explícitamente, siempre entrará como usuario anónimo, como se puede ver a continuación:

```
[root@pasarela12 ~]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
```

```
220-You are user number 1 of 2 allowed.
220-Local time is now 15:51. Server port: 21.
220-Only anonymous FTP is allowed here
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 2 minutes of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): root
230-Your bandwidth usage is restricted
230-Bienvenidos a este Servidor, anónimo.
230 Anonymous user logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
227 Entering Passive Mode (172,16,1,6,228,164)
150 Accepted data connection
drwxr-xr-x  2 0  0  4096 Apr 18 15:41 Asignaturas
-rw-r--r--  1 0  0  27 Apr 13 12:07 host.conf
226-Options: -l
226 2 matches total
ftp>
```

Para la configuración del acceso como usuario anónimo se deben cumplir las siguientes condiciones:

1. Limitar el número de conexiones máximas.

Se usa la opción **-c <NÚMERO_DE_CONEXIONES>**, que permite un máximo de conexiones o clientes conectados al servidor FTP. Por ejemplo, con **-c 42** se limitará el número máximo de accesos simultáneos a 42 clientes/conexiones. Por defecto se limita a 50.

Si indicamos la opción **-c 2**, sólo podrán estar conectados simultáneamente 2 usuarios al servidor FTP. El primer y segundo usuario entrarán sin problema, recibiendo unos mensajes de entrada como los siguientes (se entra con la IP 172.16.6.1 para mostrar que es indiferente usar esta interfaz de red o la que tiene la IP 172.16.1.6):

```
[root@pasarela12 ~]# ftp 172.16.6.1
Connected to 172.16.6.1.
220----- Welcome to Pure-FTPd [privsep] -----
```

```
220-You are user number 1 of 2 allowed.
220-Local time is now 15:54. Server port: 21.
220-Only anonymous FTP is allowed here
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 2 minutes of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.6.1:root): anonymous
230-Your bandwidth usage is restricted
230-Bienvenidos a este Servidor, anónimo.
230 Anonymous user logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
227 Entering Passive Mode (172,16,6,1,60,182)
150 Accepted data connection
drwxr-xr-x  2 0   0   4096 Apr 18 15:41 Asignaturas
-rw-r--r--  1 0   0   27 Apr 13 12:07 host.conf
226-Options: -l
226 2 matches total
ftp>
```

En la siguiente captura puede apreciarse claramente, que al entrar el tercer usuario, éste es rechazado porque no se admiten más de 2 usuarios conectados, en este caso.

```
[root@pasarela12 ~]# ftp 172.16.1.6
Connected to 172.16.1.6.
421----- Welcome to Pure-FTPd [privsep] -----
421 2 users (the maximum) are already logged in, sorry
ftp>
```

2. Presentación de mensajes de especificación de contenidos de los distintos directorios.

En cada directorio puede ponerse un fichero **.message** de forma que cada vez que se accede al directorio que lo contiene se mostrará automáticamente un mensaje, que no es más que su contenido. Si el contenido de dicho fichero indica la especificación de contenidos del directorio, tendremos el mensaje deseado. Este mensaje puede ponerse incluso en el directorio inicial del ftp,

pero al hacer el login no se mostrará, si bien, cuando nos hayamos movido por el árbol de directorios posteriormente, si accedemos de nuevo a dicho directorio, sí se mostrará el mensaje.

Aunque en apartados posteriores se verá, a continuación se muestra un fragmento de acceso a un directorio, tras lo cual se recibe un mensaje, que se a colocado a tal efecto dentro de un fichero **.message** dentro de tal directorio.

```
ftp> cd Asignaturas
250-Estas son las asignaturas que tienes este año.
250 OK. Current directory is /Asignaturas
```

3. No permitir que pueda subir archivos al FTP, pero que sí pueda descargar los ya existentes.

Se consigue con el comando **-i**, que no permite que el usuario anónimo pueda subir (upload) nada al FTP. No obstante, en una conexión anónima de FTP el usuario anónimo no tiene permiso para subir nada, y sólo tiene permiso para descargar los ficheros a los que tiene acceso. Por este motivo, en realidad, no es necesario indicar parámetro alguno.

Se ha incluido este apartado, que difiere del disponible en el guión de la práctica, porque este último no es posible realizarlo mediante el servidor FTP del que estamos haciendo uso, es decir, no podemos realizarlo con el PureFTP.

4. Limitar el número de conexiones por dirección IP.

Se consigue con el comando **-C <NÚMERO_MÁXIMO_DE_CONEXIONES_POR_IP>**, que limita el número de conexiones simultáneas que provengan de una misma dirección IP. Sólo funciona cuando el servidor no es lanzado por ningún super-servidor, pues si es lanzado por un super-servidor se supone que esta tarea ya es realizada, es decir, previene la denegación de servicios y ancho de banda como consecuencia de las acciones mal intencionadas de un único usuario.

Si lanzamos el servidor con **-C 2** no se limitará el total de conexiones a 2, sino que un mismo cliente no podrá crear más de 2 conexiones simultáneas con la misma IP. Esta característica requiere de cierta cantidad de memoria para el seguimiento o almacenamiento de direcciones IP, si bien su uso es recomendable, por razones obvias.

Adicionalmente se dispone del comando siguiente: **-y <Nº_LOGINS_MÁXIMO>:<Nº_LOGINS_ANÓNIMOS_MÁXIMO>**. Esta opción sólo funciona si el servidor se ha compilado con la opción **--with-peruserlimits**. Permite restringir el número de sesiones simultáneas que puede hacer un mismo usuario. El valor 0 indicaría que dicho número de sesiones es ilimitado.

A la hora de probar la opción **-C 1**, que permitirá que no se conecten más de 1 usuario con la misma dirección IP, como indica el comando, se deberá tener en cuenta la siguiente apreciación sobre el esta opción: Sólo funciona cuando el servidor FTP se lanza manualmente (modo standalone), es decir, si su usa un superservidor o superdemonio, se supone que éste lo hará, pero resulta que en el caso del xinetd no ocurre así.

Por este motivo, deberemos seguir el siguiente proceso para lanzar el servidor FTP de forma manual, con la opción **-C 1**, desde el equipo que tenga instalado el servidor FTP. En primer lugar matamos o detenemos el superdemonio **xinetd**:

```
service xinetd stop' o
```

Seguidamente lanzamos manualmente el demonio del servidor FTP, que en nuestro caso es el **pure-ftpd**. Para ello se debe hacer uso de una opción creada a tal efecto, que permite el lanzamiento manual o standalone. Dicho comando es **-S IP,Puerto**. Como se observa, dicho comando se acompaña de la dirección IP del servidor IP y del puerto (separados por coma). Para indicar el puerto, pueden usarse mnemotécnicos en lugar de número, en el caso de los puertos dedicados a protocolos standard, como es el caso del puerto 21 del FTP, de modo que puede ponerse **ftp**. En nuestro caso la opción se usará de la siguiente forma:

```
-S 172.16.1.6,ftp
```

Adicionalmente, hay que lanzar el demonio **pure-ftpd** en segundo plano, que si no se hace con **&**, puede hacerse con la opción **-B**, que admite el demonio del **Pure-FTP**.

Finalmente, el comando resultante, que se usará para lanzar el servidor FTP en modo standalone, será:

```
[root@pasarela12 sbin]# pure-ftpd -S 172.16.6.1,ftp -B -e -c 2 -C 1 -I 2 -A -k 1 -t 5  
[1] 4445
```

En este caso hacemos uso de la interfaz de la red interna (172.16.6.1) para lanzar el servidor FTP, en lugar de la externa. A continuación podemos comprobar que el comando **-C 1** surte efecto, intentando hacer más de una conexión desde el equipo interno a la red, cuya IP es la 172.16.6.2. La primera conexión desde esta dirección IP es aceptada:

```
[root@pasarela12 sbin]# ftp 172.16.6.1  
Connected to 172.16.6.1.
```

```
220----- Welcome to Pure-FTPd [privsep] -----  
220-You are user number 1 of 2 allowed.  
220-Local time is now 16:50. Server port: 21.  
220-Only anonymous FTP is allowed here  
220-IPv6 connections are also welcome on this server.  
220 You will be disconnected after 2 minutes of inactivity.  
500 This security scheme is not implemented  
500 This security scheme is not implemented  
KERBEROS_V4 rejected as an authentication type  
Name (172.16.6.1:root): anonymous  
230-Your bandwidth usage is restricted  
230-Bienvenidos a este Servidor, anónimo.  
230 Anonymous user logged in  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp> ls  
227 Entering Passive Mode (172,16,6,1,153,75)  
150 Accepted data connection  
drwxr-xr-x  2 0    0    4096 Apr 18 15:41 Asignaturas  
-rwxr-xr-x  1 0    0    206592 Apr  6 10:53 ejemplo.rpm  
-rw-r--r--  1 0    0    27 Apr 13 12:07 host.conf  
226-Options: -l  
226 3 matches total  
ftp>
```

Sin embargo, la segunda conexión desde la misma IP es rechazada, pues sólo se permite una:

```
[root@pasarela12 ~]# ftp 172.16.6.1  
Connected to 172.16.6.1.  
421 Too many connections (1) from this IP  
ftp>
```

Al usar el modo standalone, para el lanzamiento del servidor FTP, a la hora de conectarse con él hay que indicar exactamente la IP que se indicó en la opción **-S**, a la hora de conectarse a él como cliente. Esto quiere decir que sólo funcionará la conexión si usamos **ftp 172.16.6.1**, mientras que con **ftp 172.16.6.2** fallará.

5. Desconexión en caso de inactividad.

Por comandos se hace uso de la opción **-I <MINUTOS>**. Así si en la opción `server_args` ponemos **-I 1** se echará al cliente transcurrido 1 minuto de inactividad. Que es idéntico a lo especificado en la sección **Acceso en modo usuario del sistema**.

Aunque ya se vio previamente, ahora fijamos en 2 minutos el periodo de inactividad, con la opción **-I 2**, de forma que obtenemos:

```
[root@pasarela12 ~]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
220-You are user number 1 of 2 allowed.
220-Local time is now 15:53. Server port: 21.
220-Only anonymous FTP is allowed here
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 2 minutes of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): anonymous
230-Your bandwidth usage is restricted
230-Bienvenidos a este Servidor, anónimo.
230 Anonymous user logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
421 Timeout - try typing a little faster next time
Passive mode refused. Turning off passive mode.
No control connection for command: Transport endpoint is not connected
ftp>
```

6. Los usuarios no puedan salir de los directorios de los cuales son propietarios (chroot).

Con la opción **-a <GID>** se indica el **GID** indica el grupo de usuario que tiene acceso ilimitado por todo el árbol del sistema de ficheros, es decir, que puede salir fuera de su directorio personal. De este modo se limita a los miembros de un grupo la posibilidad de salir de su directorio, mientras que el resto no puede salir. Se conoce como **chroot** al hecho por el cual un usuario puede acceder a su directorio personal pero no puede acceder o salir fuera del mismo. No obstante, el

usuario **root** (con uid 0) siempre tiene acceso total a todo el sistema de ficheros (incluso aunque esté fuera del grupo gid que se indique con la opción **-a**).

Como lo que nosotros pretendemos es que ningún usuario (salvo el **root**) puede salir de su directorio personal, también podemos hacer uso de una opción más cómoda, que es **-A**, que hace esto directamente, por lo que de hecho es la que usaremos. En el siguiente ejemplo se observa como no puede accederse a directorios externos a nuestra zona a jaula (directorio en el que estamos enjaulados: **chroot**) y ni siquiera listar el contenido de los mismo con el comando **ls**.

```
[root@pasarela12 ~]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
220-You are user number 1 of 2 allowed.
220-Local time is now 16:10. Server port: 21.
220-Only anonymous FTP is allowed here
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 2 minutes of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): anonymous
230-Your bandwidth usage is restricted
230-Bienvenidos a este Servidor, anónimo.
230 Anonymous user logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd ..
250 OK. Current directory is /
ftp> ls
227 Entering Passive Mode (172,16,1,6,87,195)
150 Accepted data connection
drwxr-xr-x  2 0    0      4096 Apr 18 15:41 Asignaturas
-rw-r--r--  1 0    0      27 Apr 13 12:07 host.conf
226-Options: -l
226 2 matches total
ftp>
ftp> ls ..
227 Entering Passive Mode (172,16,1,6,215,12)
```

```
150 Accepted data connection
drwxr-xr-x  2 0    0      4096 Apr 18 15:41 Asignaturas
-rw-r--r--  1 0    0      27 Apr 13 12:07 host.conf
226-Options: -l
226 2 matches total
```

Al hace `cd ..` se dice que el directorio actual es / pero en realidad seguimos en el directorio en que estábamos, es decir, no salimos del directorio en que estamos enjaulado. Podemos hacer `ls`, pero no `ls ..`, pues el segundo intenta mostrar el contenido del directorio superior, al cual no tenemos acceso. El comando `ls` a secas indica que sí tenemos acceso al contenido de nuestro directorio o jaula. Igualmente, podemos acceder a las carpetas del mismo:

```
ftp> cd Asignaturas
250-Estas son las asignaturas que tienes este año.
250 OK. Current directory is /Asignaturas
ftp> ls
227 Entering Passive Mode (172,16,1,6,121,252)
150 Accepted data connection
-rw-----  1 0    0      0 Apr 18 15:34 Geometria
-rw-----  1 0    0      0 Apr 18 15:33 Mates
226-Options: -l
226 2 matches total
ftp>
```

7. Presentación de un mensaje de bienvenida cuando un usuario acceda al ftp.

Para que aparezca un mensaje de bienvenida tras logearse en el ftp, deberemos poner un fichero con el nombre **.banner** en la carpeta raíz o carpeta inicial que se abre cuando el usuario se conecta al ftp. De este modo se mostrará su contenido.

El directorio de inicio variará en función del usuario que entre al servidor FTP y de forma general depende del que se haya asignado al usuario. Los caso principales son:

- a) Superusuario o root. Su carpeta es **/root**, a no ser que se la cambie.
- b) Usuario del sistema. Por lo general, salvo que no se cambie, usará la carpeta **/home/USUARIO**, es decir, **/home** y una carpeta con su nombre.
- c) Usuario anónimo (anonymous). Por lo general, el sistema reserva la carpeta **/var/ftp**, para los usuarios anónimos.

Esto ya fue comentado en el apartado **Acceso en modo usuario del sistema**, junto con los mensajes por directorios y las cookies. En este caso estamos tratando al usuario anónimo, de modo que deberemos crear el fichero **.banner** en el fichero **/var/ftp**.

Como ejemplo, en todas las capturas realizadas en las que se conecta con el servidor FTP se recibe algún mensaje de bienvenida, cuyo valor se extrae del fichero **.banner** conectado en el directorio raíz o personal del usuario que se conecte. A continuación se muestra dicho fragmento, de alguno de los casos:

```
[root@pasarela12 ~]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
220-You are user number 1 of 2 allowed.
220-Local time is now 16:15. Server port: 21.
220-Only anonymous FTP is allowed here
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 2 minutes of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): anonymous
230-Your bandwidth usage is restricted
230-Bienvenidos a este Servidor, anónimo.
```

8. Limitar el tamaño utilizado de la partición.

Se usa la opción **-k <PORCENTAJE>**, de forma que no permite subir más archivos al servidor FTP si la partición del usuario (en este caso el usuario anónimo) tiene más del **PORCENTAJE** de ocupación de la misma, para llegar a estar llena. Por ejemplo, con **-k 90** conseguiremos que el disco nunca se llene más del 90 % de su capacidad, a través del FTP.

Haciendo uso de la opción **-k 1**, el tamaño de disco del que disponemos será de un 1% del total de la partición; por lo general es un tamaño tan pequeño que no podremos subir nada al servidor FTP. Con ello se pretende probar que la opción funciona correctamente, como se ve en el siguiente caso:

```
[root@pasarela12 ~]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
```

```
220-You are user number 1 of 2 allowed.
220-Local time is now 16:15. Server port: 21.
220-Only anonymous FTP is allowed here
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 2 minutes of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): anonymous
230-Your bandwidth usage is restricted
230-Bienvenidos a este Servidor, anónimo.
230 Anonymous user logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
227 Entering Passive Mode (172,16,1,6,38,194)
150 Accepted data connection
drwxr-xr-x  2 0   0   4096 Apr 18 15:41 Asignaturas
-rw-r--r--  1 0   0   27 Apr 13 12:07 host.conf
226-Options: -l
226 2 matches total
ftp> put ejemplo.tar.gz
local: ejemplo.tar.gz remote: ejemplo.tar.gz
227 Entering Passive Mode (172,16,1,6,111,98)
553 Disk full - please upload later
ftp>
```

El mensaje de **Disk full** (disco lleno) indica que la partición tiene más de un 1% llena y por ello no podemos subir nada.

9. Limitar el ancho de banda.

Mediante comandos se usa la opción siguiente: **-T <Kbytes/Segundo> :** **<Kbytes/Segundo>** para el ancho de banda de subida y de bajada, respectivamente. Si se usa el comando **-t** en lugar de **-T** la semántica de limitación de ancho varía de la siguiente forma:

-t --> Limitación del ancho de banda para el usuario anónimo en concreto.

-T --> Limitación del ancho de banda para todos los usuarios.

En conclusión, lo recomendable es usar la opción **-t <Kbytes/Segundo>** : **<Kbytes/Segundo>**.

En modo estático, se usa la opción **UserBandwidth <Kbytes/Segundo>**, que limitará el ancho de banda para todos los usuarios incluyendo al usuario anónimo. Esta velocidad incluirá a la velocidad de subida y de bajada conjuntamente. Y la opción **AnonymousBandwidth <Kbytes/Segundo>** permite que la limitación afecte exclusivamente al usuario anónimo.

Fijando la opción **-t 5**, la velocidad de subida y bajada se limitará a 5Kb/seg. Haremos uso de un fichero llamado **f** para subirlo, si bien no podremos porque se mantiene la opción **-k 1**, del apartado anterior, que limita el uso de la partición a un 1% (que ya está ocupado). También se hará uso de otro fichero ya presente en el servidor FTP, cuyo nombre es **ejemplo.rpm**, de un cierto tamaño, para descargarlo.

```
[root@pasarela12 ~]# ftp 172.16.1.6
Connected to 172.16.1.6.
220----- Welcome to Pure-FTPd [privsep] -----
220-You are user number 1 of 2 allowed.
220-Local time is now 16:19. Server port: 21.
220-Only anonymous FTP is allowed here
220-IPv6 connections are also welcome on this server.
220 You will be disconnected after 2 minutes of inactivity.
500 This security scheme is not implemented
500 This security scheme is not implemented
KERBEROS_V4 rejected as an authentication type
Name (172.16.1.6:root): anonymous
230-Your bandwidth usage is restricted
230-Bienvenidos a este Servidor, anónimo.
230 Anonymous user logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
227 Entering Passive Mode (172,16,1,6,218,87)
150 Accepted data connection
drwxr-xr-x  2 0  0    4096 Apr 18 15:41 Asignaturas
-rwxr-xr-x  1 0  0   206592 Apr  6 10:53 ejemplo.rpm
-rw-r--r--  1 0  0    27 Apr 13 12:07 host.conf
226-Options: -l
226 3 matches total
```

```
ftp> put f
local: f remote: f
227 Entering Passive Mode (172,16,1,6,73,98)
553 Disk full - please upload later
ftp> ls
227 Entering Passive Mode (172,16,1,6,84,83)
150 Accepted data connection
drwxr-xr-x  2 0  0    4096 Apr 18 15:41 Asignaturas
-rwxr-xr-x  1 0  0   206592 Apr  6 10:53 ejemplo.rpm
-rw-r--r--  1 0  0    27 Apr 13 12:07 host.conf
226-Options: -l
226 3 matches total
ftp> get ejemplo.rpm
local: ejemplo.rpm remote: ejemplo.rpm
227 Entering Passive Mode (172,16,1,6,76,198)
150-Accepted data connection
150 201.8 kbytes to download
226-File successfully transferred
226 40.351 seconds (measured here), 5.00 Kbytes per second
206592 bytes received in 40 seconds (5 Kbytes/s)
ftp>
```

Las posibles opciones que se terminen por usar finalmente serían:

```
server_args = -e -c 2 -C 1 -I 2 -A -k 1 -t 5
```

Dichas opciones irán en el campo **server_args** dentro del fichero de configuración del superdemonio **xinetd**.

Acceso como usuarios virtuales

Los usuarios virtuales son un mecanismo simple para almacenar una lista de usuarios con su contraseña, nombre, uid, directorio por defecto, etc. Son algo parecido a **/etc/passwd**. Pero no usan dicho fichero, es uno diferente exclusivamente para el FTP.

Su objetivo es separar los usuarios de FTP de los usuarios del sistema. Además, cada usuario virtual puede tener unos parámetros diferentes en lo referente a cuotas de espacio, ancho de banda,

ratios, etc.

Antes de crear usuarios virtuales, deberemos asignarlos a un usuario del sistema, por lo que es muy recomendable crear uno exclusivamente para estos menesteres. En nuestro caso crearemos el usuario **ftpuser** y el grupo **ftpgroup** que usaremos exclusivamente para el FTP. Además, dicho usuario no será usable como usuario de acceso al sistema, por lo cual no le asignamos directorio por defecto ni shell.

```
groupadd ftpgroup  
useradd -g ftpgroup -d /dev/null -s /dev/null ftpuser
```

A partir de ahora, toda la configuración de los usuarios virtuales la haremos con la propia utilidad del Pure-FTPd: **pure-pw**. También se podrían editar los ficheros a mano, ya que las contraseñas son compatibles con las funciones de hash que usa el sistema, sin embargo el uso de la utilidad que trae el programa simplificaría el proceso. El formato del archivo de usuarios de FTP es el siguiente:

```
<account>:<password>:<uid>:<gid>:<gecos>:<home directory>:<upload  
bandwidth>:<download bandwidth>:<upload ratio>:<download ratio>:<max number  
of connections>:<files quota>:<size quota>:<authorized local IPs>:<refused  
local IPs>:<authorized client IPs>:<refused client IPs>:<time  
restrictions>
```

Aceptando usuarios virtuales

Para que el servidor admita usuarios virtuales, deberemos especificarle en donde se encuentra el archivo binario de configuración mediante la opción:

```
-lpuredb:<archivo_pdb>
```

El archivo donde haremos los cambios inicialmente será un archivo en modo texto, generalmente con extensión **.passwd**. Sin embargo, el archivo final que usa el servidor es un archivo binario, generalmente con extensión **.pdb**, que es el que tendremos que especificar. En la sección **Aplicando los cambios** explicaremos con más detalle como pasar de un formato a otro.

Creando un usuario

Para crear un usuario, usamos la siguiente sintaxis:

```
pure-pw useradd <login> [-f <passwd file>] -u <uid> [-g <gid>]
-D/-d <home directory> [-c <gecos>]
[-t <download bandwidth>] [-T <upload bandwidth>]
[-n <max number of files>] [-N <max Mbytes>]
[-q <upload ratio>] [-Q <download ratio>]
[-r <allow client host>[/<mask>]][,<allow client host>[/<mask>]]...]
[-R <deny client host>[/<mask>]][,<deny client host>[/<mask>]]...]
[-i <allow local host>[/<mask>]][,<allow client host>[/<mask>]]...]
[-I <deny local host>[/<mask>]][,<deny local host>[/<mask>]]...]
[-y <max number of concurrent sessions>]
[-z <hhmm>-<hhmm>] [-m]
```

A continuación creamos el usuario **Joe**, cuyo directorio por defecto será **/home/ftpuser/joe** y el usuario del sistema asociado será **ftpuser**:

```
pure-pw useradd joe -u ftpuser -d /home/ftpuser/joe -j
```

La opción **-d** hace que joe no pueda salir de su directorio por defecto (chrooted). En caso de querer dejarle salir, usaríamos la opción **-D**. Además, la opción **-j** crea automáticamente el directorio por defecto para no tener que crearlo manualmente.

Cambiando opciones de los usuarios

Una vez que el usuario es creado, se puede editar su información y cambiar sus opciones. Para ello usaremos el comando **pure-pw usermod**, cuyas opciones son iguales al ya visto **pure-pw useradd**. Por ejemplo, para limitar la cuota de joe a 10000 ficheros y 10 Megabytes sería:

```
pure-pw usermod joe -n 1000 -N 10
```

En el caso de querer quitar las cuotas, usaríamos:

```
use pure-pw usermod joe -n -N
```

Si lo que queremos es cambiar la contraseña de joe, podemos usar:

```
pure-pw passwd joe [-f <archivo_pass>]
```

donde la opción **-f** indica el archivo de usuarios de ftp que estamos usando en caso de no ser

el predeterminado.

Mostrando y borarando usuarios

La sintaxis para mostrar un usuario es:

```
pure-pw show <login> [-f <passwd file>]
```

Para mostrar la información que hay acerca de joe, usaremos:

```
pure-pw show joe
```

La sintaxis para borrar un usuario es:

```
pure-pw userdel <login> [-f <passwd file>]
```

Para borrar al usuario joe usaremos:

```
pure-pw userdel joe
```

Hay que tener en cuenta que la información que contiene su directorio por defecto se mantiene, teniendo que borrarla manualmente si lo deseamos.

Aplicando los cambios

Si hemos realizado cambios en la configuración, el servidor no las aplica directamente. Para ello deberemos transformar el archivo de opciones que tenemos, generalmente **/etc/pureftpd.passwd** a otro archivo con un formato binario de acceso rápido, generalmente **/etc/pureftpd.pdb**. Para ello usaremos la utilidad **pure-pw mkdb**. Su sintaxis es:

```
pure-pw mkdb [<archivo_pdb>] [-f <archivo_passwd>]
```

En nuestro ejemplo, como los archivos de opciones son los por defecto, usaremos:

```
pure-pw mkdb
```

Los cambios ya han sido aplicados, sin necesidad de reiniciar el servidor. Además, existe la opción **-m** en los demás comandos, que directamente hace que los cambios sean aplicados en el servidor, y por tanto ambos ficheros son modificados (**passwd** y **pdb**). De hecho **-m** reconstruye

/etc/pureftpd.pdb automáticamente, y no se limita a simplemente actualizar **/etc/pureftpd.passwd**.

Creación de un Servidor Virtual

Crearemos un servidor virtual del dominio **ftpred6.redes6.redes.dis.ulpgc.es** (no podrá ser **ftpred6.redes.dis.ulpgc.es**, dado que nuestro dominio es **redes6.redes.dis.ulpgc.es** y no tenemos acceso al dominio superior, cuyo nombre es **redes.dis.ulpgc.es**) con acceso únicamente en modo anónimo. Para ello el proceso a seguir consta de los siguientes pasos:

1. Crear una nueva ip para la interfaz que está de cara a la red internet, es decir, la interfaz **eth1**, cuya IP es **172.16.6.1**. Este proceso es absolutamente necesario para luego poder realizar la creación de uno o más servidores virtuales. Haciéndolo de forma estática, se obtendrá un fichero de configuración de la interfaz en **/etc/sysconfig/network-scripts/ifcfg-eth1:1**, cuyo contenido es el siguiente:

```
DEVICE=eth1:1
BOOTPROTO=static
ONBOOT=yes
TYPE=Ethernet
IPADDR=172.16.6.124
NETWORK=172.16.6.0
NETMASK=255.255.255.0
USERCTL=no
PEERDNS=yes
IPV6INIT=no
HWADDR=00:c0:49:b2:be:a9
GATEWAY=172.16.1.6
```

Ahora es como si tuviéramos una nueva interfaz de red, que podríamos llamar **eth1:1**, que será la usada para el Servidor Virtual, cuya dirección IP es la **172.16.6.124**, visible desde la red interna.

2. Crear un directorio destinado a almacenar los ficheros del Servidor Virtual. En nuestro caso dicho directorio será el siguiente: **/home/customers/hola/ftp**. Se hará con el comando siguiente:

```
mkdir /home/customers/hola/ftp
```

3. Crear el enlace simbólico (link) entre el directorio anterior y la IP de la interfaz de red que se usará para el Servidor Virtual, es decir, la **172.16.6.124**, que se ubicará en el

directorio del **Pure-FTP**, que es el siguiente: **/etc/pure-ftpd**. Desde el directorio **/etc/pure-ftpd**, se usará el comando de creación de enlaces, con el formato **ln -s DESTINO ENLACE**. El proceso a base de comandos es:

```
[root@pasarela12 hola]# cd /etc/pure-ftpd
[root@pasarela12 pure-ftpd]# ln -s /home/customers/hola/ftp 172.16.6.124
[root@pasarela12 pure-ftpd]# ls
172.16.6.124
```

4. El siguiente paso consiste en la correcta configuración del servidor DNS para que permita resolver el nombre del Servidor Virtual a crear, con el nombre **ftpred6.redes6.redes.dis.ulpgc.es**, al que le corresponderá la dirección IP 172.16.6.124, de la interfaz de red **eth1:1**, creada a tal efecto. Esto conlleva las siguientes modificaciones:

- i. Configuración del nuevo registro para el nombre **ftpred6.redes6.redes.dis.ulpgc.es**, con la dirección IP 172.16.6.124. Esto se realiza en el fichero **db.redes6**, que ya existirá, creado originalmente para la práctica 2, consistente en la puesta en marcha de un servidor DNS. El fichero resultante tendrá el siguiente aspecto:

```
// Valor recomendado de TTL (evita un warning, pues si no se pone usa el ttl mínimo)
// El warning es: no TTL specified; using SOA MINTTL instead (en la línea 1)
$TTL 86400
redes6.redes.dis.ulpgc.es. IN SOA davidj.redes6.redes.dis.ulpgc.es. root.redes6.redes.dis.ulpgc.es. (
    2002032006      ;Serial
    28800           ;Refresco
    14400           ;reintento
    3600000         ;Expira
    86400           ;ttl minimo
)

    IN NS  davidj.redes6.redes.dis.ulpgc.es.  ;DNS primario
    IN NS  enrique.redes6.redes.dis.ulpgc.es. ;DNS secundario
    IN NS  rodrigo.redes2.redes.dis.ulpgc.es. ;DNS secundario externo

localhost    IN A    127.0.0.1
davidj       IN A    172.16.1.6
davidj       IN A    172.16.6.1
```

```
enrique      IN A    172.16.6.2  
ftpred6     IN A    172.16.6.124
```

La única línea modificada es la última:

```
ftpred6     IN A    172.16.6.124
```

Con esta línea se indica que el nombre canónico **ftpred6.redes6.redes.dis.ulpgc.es**, que se corresponde con la dirección IP 172.16.6.124. Esto permitirá posteriormente las conexiones FTP con dicho Servidor Virtual, mediante el comando **ftp ftpred6.redes6.redes.dis.ulpgc.es** directamente.

ii. Adicionalmente, para que se haga uso del servidor DNS nuestro, deberemos indicarlo en el fichero **/etc/resolv.conf**, de forma que éste quedará de la siguiente forma:

```
search redes6.redes.dis.ulpgc.es  
nameserver 172.16.6.1
```

5. Finalmente, se procede a la reinicialización de todos los servicios afectados, que son los siguientes: **network**, **xinetd** y **named**. Por ello, ejecutaremos:

```
service network restart; service xinetd restart, service named restart
```

6. Adicionalmente podemos crear un fichero **.banner** que se situará en el directorio de entrada de los usuarios que accedan al Servidor Virtual, que será el directorio **/home/customers/hola/ftp**. Dicho fichero **.banner** contendrá:

```
Hola. Bienvenido al servidor virtual.
```

Además de estas configuraciones de puesta en marcha del Servidor Virtual, en la configuración de los argumentos del superdemonio **xinetd** se indicará la ya conocida opción **-e**, para que el acceso al servidor sólo sea en modo anónimo. Como mínimo se tendrán los siguientes parámetros:

```
server_args = -e
```

Una vez realizados los pasos de configuración, procedemos a realizar un acceso al Servidor Virtual, de forma que se obtiene:

```
[root@pasarela12 named]# ftp ftpred6.redes6.redes.dis.ulpgc.es  
Connected to ftpred6.redes6.redes.dis.ulpgc.es.
```

```
220----- Welcome to Pure-FTPd [privsep] -----  
220-You are user number 1 of 50 allowed.  
220-Local time is now 18:29. Server port: 21.  
220-IPv6 connections are also welcome on this server.  
220 You will be disconnected after 15 minutes of inactivity.  
500 This security scheme is not implemented  
500 This security scheme is not implemented  
KERBEROS_V4 rejected as an authentication type  
Name (hola:root): anonymous  
230-Hola. Bienvenido al servidor virtual.  
230 Anonymous user logged in  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp> ls  
227 Entering Passive Mode (172,16,6,124,25,22)  
150 Accepted data connection  
-rw-r--r--  1 0    0      0 Apr 15 17:19 hola  
-rw-r--r--  1 0    0      0 Apr 15 17:19 ola  
226-Options: -l  
226 2 matches total  
ftp>
```

También podría accederse poniendo simplemente **ftp ftpred6**, ya que estamos dentro del dominio **redes6.redes.dis.ulpgc.es**. Se observa como sólo se permite el acceso en modo anónimo y como se muestra el mensaje de bienvenida comentado con anterioridad. Igualmente, se muestra algunos de los ficheros colocados en la carpeta de entrada por defecto, que se encuentra en **/home/customers/hola/ftp**, y que contiene los ficheros **hola** y **ola**, como se ve en la captura anterior.