# Resumen Tema 4.

# Instalación y administración de servidores FTP.

## 1.- Servicio de transferencia de ficheros

Uno de los métodos más empleados como servicio de transferencia de archivos se realiza mediante el servicio ftp.

Utiliza FTP empleando la arquitectura cliente-servidor.

Principal problema del protocolo FTP es la no seguridad de la información, esto es, la transferencia tiene lugar sin cifrar la información transferida.

Una mejora es FTPS empleando cifrado SSL/TLS.

NO confundir FTPS con SFTP, ya que este último es implementado con otro servicio, el servicio SSH, y es utilizado para conexiones remotas seguras a través de un terminal de comandos.

## 1.1.- ¿Cómo funciona?

El protocolo FTP emplea una arquitectura cliente/servidor siendo el cliente ftp quien solicita la transferencia de archivos y el servidor ftp quien ofrece los archivos.

Pertenece a la familia de protocolos de red TCP y orientado a la conexión. Sino se establece la conexión ésta no tiene lugar.

FTP no utiliza una autenticación cifrada y se considera un protocolo inseguro. Pero muy extentido.

FTPS es otro protocolo con comunicaciones cifradas.

FTP requiere de dos puertos TCP, un puerto es necesario para establecer el control de la conexión (cliente-servidor) y el otro es para el control de la transmisión (datos).

Los puertos TCP suelen ser por defecto el 21 para el control de conexión (comandos) y otro por ejemplo el 20 o mayor de 1024 para datos.

La ventaja que supone utilizar el protocolo FTP se basa en su alto rendimiento y sencillez, que lo hacen una opción conveniente para la transferencia de archivos.

## 1.2.- Cliente FTP

Para poder establecer una conexión con el protocolo FTP son necesarias dos partes: un servidor y un cliente.

Clientes ftp: terminal de comandos, clientes gráficos, navegación web.

Cuál elegir, depende:

* Conoces la consola ftp? Mejor una terminal
* Cuál es el uso? Para que lo vas a utilizar? Si sólo es descargar mejor un cliente ftp navegador.
* Quieres reanudar la conexión en caso de corte en la misma? Cliente gráfico.
* Deseas facilidad de manejo? Cliente gráfico.
* Qué tipo de conexión quieres establecer? Cifrada, no cifrada, no todos los clientes soportan este tipo de conexión.
* Deseas recordar conexiones? No todos los clientes ftp lo permiten.

|  |
| --- |
| Cliente Filezilla:   * Fácil de usar. * Soporta FTP, FTP sobre SSL / TLS (FTPS) y SFTP. * Compatibilidad con múltiples plataformas: se ejecuta en Windows, Linux, BSD, Mac OS X y más. * Soporte IPv6. * Disponible en varios idiomas. * Soporta y reanuda la transferencia de archivos de gran tamaño (mayores de 4 GB). * Interfaz de usuario con pestañas. * Potente administrador de sitios y cola de transferencia. * Marcadores. * Arrastrar y soltar. * Permite configurar límites de velocidad de transferencia. * Nombre de filtros. * Directorio de comparación. * Asistente de configuración de la red. * Edición de archivos remoto. * Automantenimiento de la conexión. * HTTP/1.1, SOCKS5 y soporte de FTP-Proxy. * Fichero de registro. * Sincronización de directorios de navegación. * Búsqueda de archivos remoto. |

## 1.3.- Tipos de usuarios.

Dos tipos de usuarios:

* Usuarios anónimos: usuarios que tiene acceso y permisos limitados
* Usuarios del sistema: disponen de una cuenta en la máquina que ofrece el servicio FTP. Necesitan un usuario y contraseña.

Pero en ciertos servidores, como el servidor ProFTPD, existe una tercera posibilidad muy interesante: **usuarios virtuales**. Los usuarios virtuales poseen acceso y permisos al servidor FTP sin necesidad de ser usuarios del sistema, por lo tanto si un usuario virtual quisiera acceder al sistema operativo como si fuese un usuario del sistema, ya sea de forma local o remota no podría, pues su cuenta de usuario no existe en el sistema. Los usuarios virtuales tienen definida una contraseña propia y pueden estar definidos en ficheros de autenticación (de texto) con el mismo formato que los del sistema operativo GNU/Linux /etc/passwd, directorios, bases de datos SQL y servidores RADIUS.

|  |
| --- |
| Dependiendo del servidor ftp, podrás tener unos métodos de autenticación de usuarios u otros,por ejemplo en el servidor ftp **ProFTPD** se permite los siguientes métodos:   * Ficheros de autenticación del sistema operativo: **/etc/passwd** y **/etc/shadow**: Para ello usa las directivas **AuthUserFile** y **AuthGroupFile**. * Usuarios virtuales definidos mediante ficheros de autenticación (de texto) propios, distintos de los del sistema operativo: para ello también usa las directivas **AuthUserFile** y **AuthGroupFile**. * Autenticación PAM : Es necesario establecer la directiva **AuthPAMAuthorative** a 'on'. * Bases de datos SQL, tales como MySQL o Postgres. Para ello emplea el módulo mod\_sql;más información sobre el uso de mod\_sql lo puedes encontrar en el HowTo SQL * LDAP: Para ello emplea el módulo mod\_ldap. * RADIUS: Para ello emplea el módulo mod\_radius.   Mediante la directiva **UserPassword** se puede crear una contraseña para un usuario particular que sobreescribe la contraseña del usuario en **/etc/passwd** (o **/etc/shadow**), esta contraseña es solamente efectiva dentro del contexto en el cual la directiva es aplicada, esto es, no se modifica el fichero **/etc/passwd** (o **/etc/shadow**) sino que se da la posibilidad de que el usuario emplee otra contraseña distinta de la definida en los ficheros del sistema operativo. |

## 1.4.- Modos de conexión del cliente.

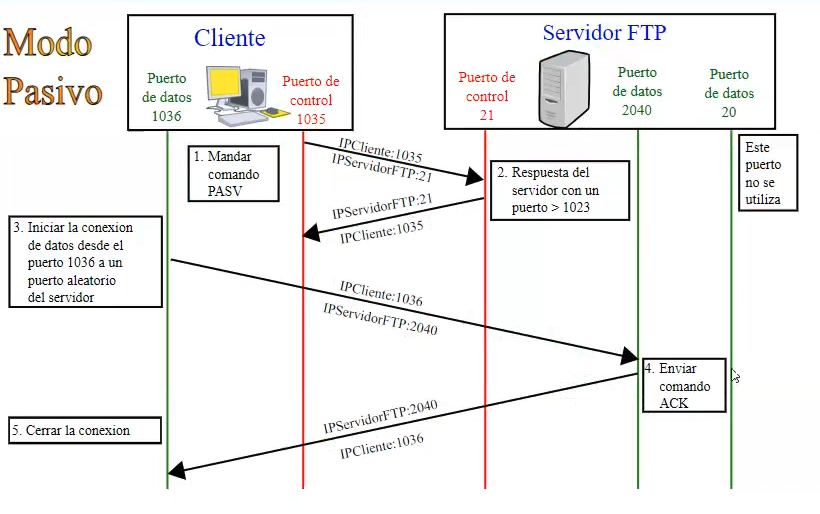
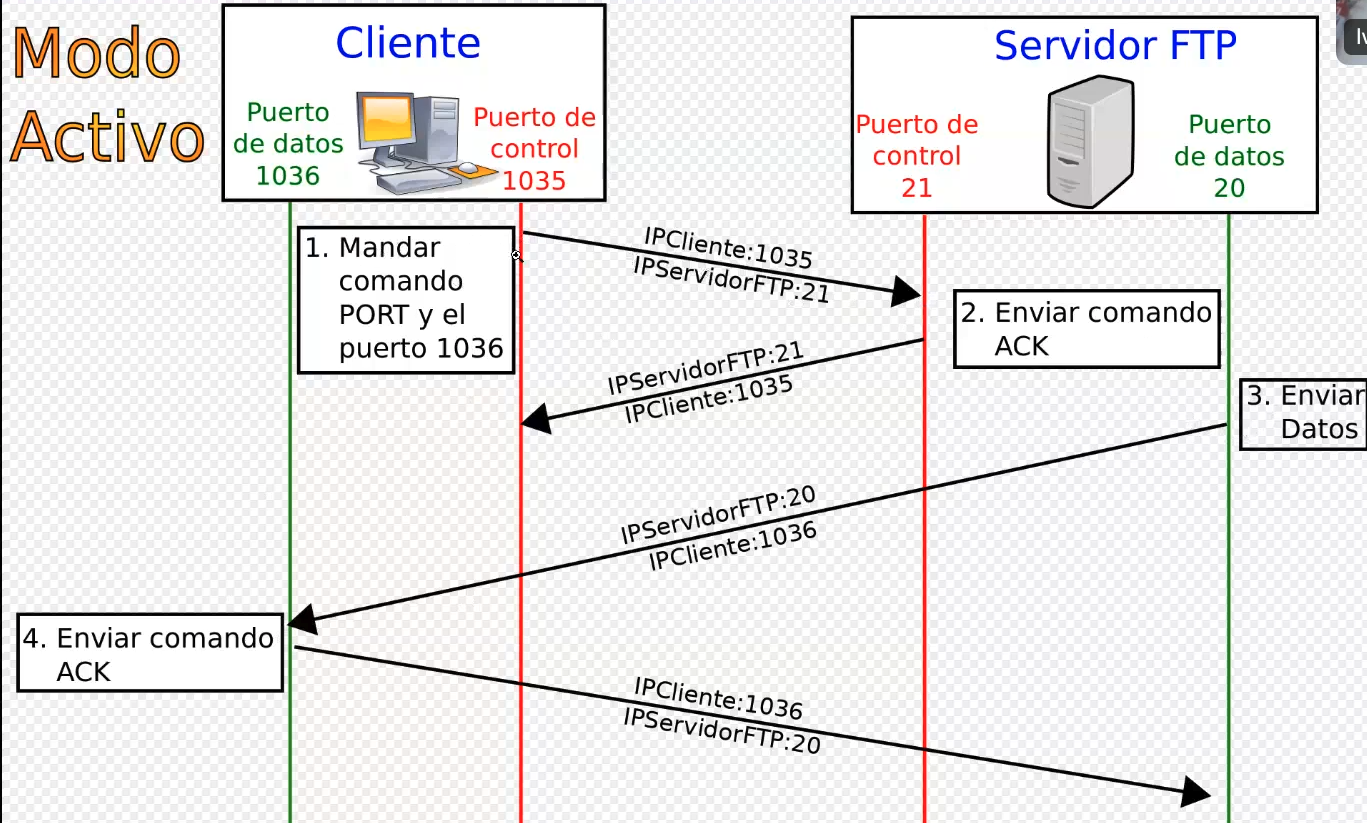
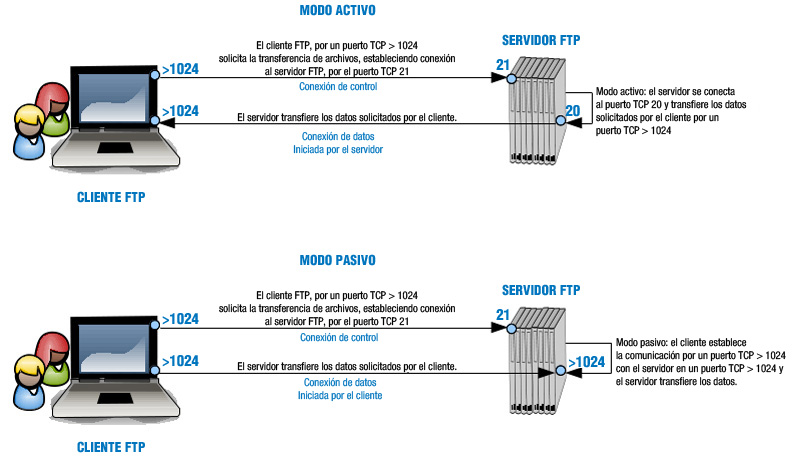
FTP necesita de dos puertos y básicamente depende de dos factores: del modo de conexión del cliente ftp y de la configuración del servidor ftp.

Por defecto el puerto **TCP 21** es el canal de conexión de control.

El puerto del canal de transmisión de datos depende del modo de conexión del cliente ftp que puede ser activo o pasivo.

* **El modo activo:** es el método original utilizado por el protocolo FTP para la transferencia de datos a la aplicación cliente. Cuando el cliente FTP inicia una transferencia de datos, el servidor abre una conexión desde el puerto 20 en el servidor para la dirección IP y un puerto aleatorio sin privilegios (mayor que 1024) especificado por el cliente. Este arreglo implica que la máquina cliente debe poder aceptar conexiones en cualquier puerto superior al 1024. Con el crecimiento de las redes inseguras, tales como Internet, es muy común el uso de cortafuegos para proteger las máquinas cliente. Debido a que estos cortafuegos en el lado del cliente normalmente rechazan las conexiones entrantes desde servidores FTP en modo activo, se creó el modo pasivo.
* La aplicación FTP cliente es la que inicia el modo pasivo, de la misma forma que el modo activo. El cliente FTP indica que desea acceder a los datos en modo pasivo y el servidor proporciona la dirección IP y el puerto aleatorio, sin privilegios (mayor que 1024) en el servidor. Luego, el cliente se conecta al puerto en el servidor y descarga la información requerida.

|  |
| --- |
| En sistemas GNU/Linux es típico encontrar el archivo **/etc/services** que contiene una lista de puertos TCP/UDP relacionado con los servicios estándar que trabajan en los mismos. Ejecuta el comando **cat /etc/services | grep ftp** y encontrarás todos los puertosy servidores relacionados con la cadena **ftp**. |



## 1.5.- Tipos de transferencia de archivos.

Se agrupan en dos tipos:

* Archivos ASCII, en texto plano .txt, .ps , .html.
* Archivos binarios: .exe, .jpg, .png, .mp3, .wav, .avi, .mov, etc

El servidor ftp permite configurar la transferencia de archivos según el tipo del mismo. Comandos: **ascii** y **binary**.

## 1.6.- Establecer permisos en ftp.

Existen 3 grupos:

* Propietario (user=u): el creador o el que ha subido el archivo al servidor FTP.
* Grupo (group=g): se refiere a un grupo de usuarios que posee la propiedad del archivo.
* Otros (others=o): son el resto de usuarios no propietarios o que no pertenecen al grupo indicado.

Cada grupo tiene tres permisos: lectura, escritura, ejecución (rwx) y – sin permisos. (r=4, w=2, x=1, -=0).

ls –l //podemos ver los permisos.

-rw-r--r-- 1 alumno clase 0 jun 20 01:15 prueba1.txt

Por lo tanto, los permisos rw-r- -r- - equivalen a 644.

En un sistema GNU/Linux, en principio, no todos los usuarios del sistema tienen acceso por ftp, así existe un fichero **/etc/ftpusers** que contiene una lista de usuarios que **no tienen permiso** de acceso por FTP. Por razones de seguridad al menos los siguientes usuarios deberían estar listados en este fichero: root, bin, uucp, news. Ten en cuenta que las líneas en blanco y las líneas que comiencen por el carácter '#' serán ignoradas.

## 1.7. Servicio de transferencia de archivos en modo texto.

Cliente en modo texto, cabe destacar dos: el cliente en modo texto ftp y el cliente en modo texto iftp.

Debian se dispone del cliente modo ftp en una instalación básica. Para poder utilizarlo en el sistema simplemente hay que ejecutarlo como comando.

1. Sintaxis:

Ftp host port

* Host: el servidor ftp
* Port: el puerto, 21 por defecto.

Man ftp o info ftp (ayuda).

2. Al ejecutar el comando se abrirá una consola propia de ftp en la cual puedes introducir comandos ftp.

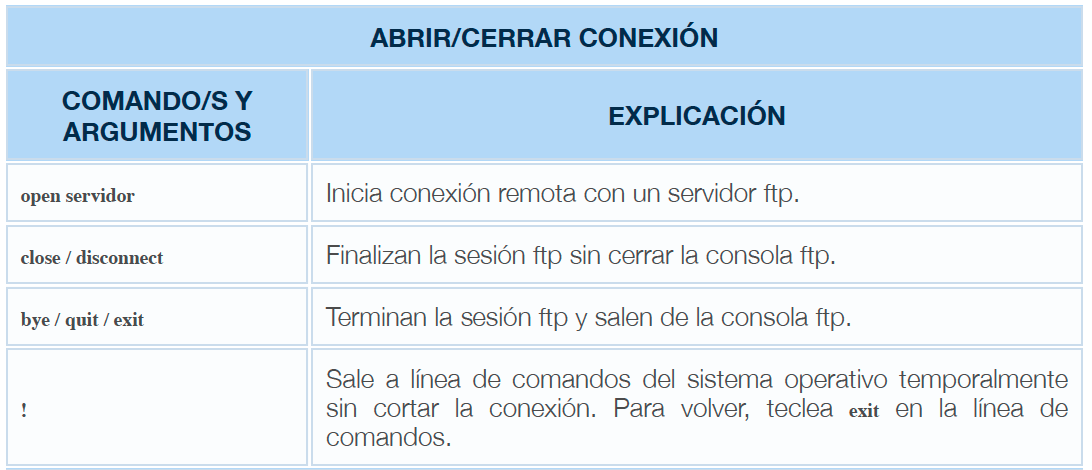
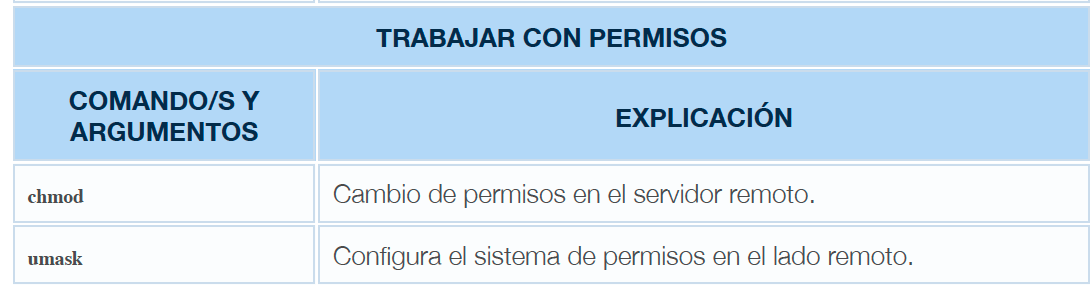
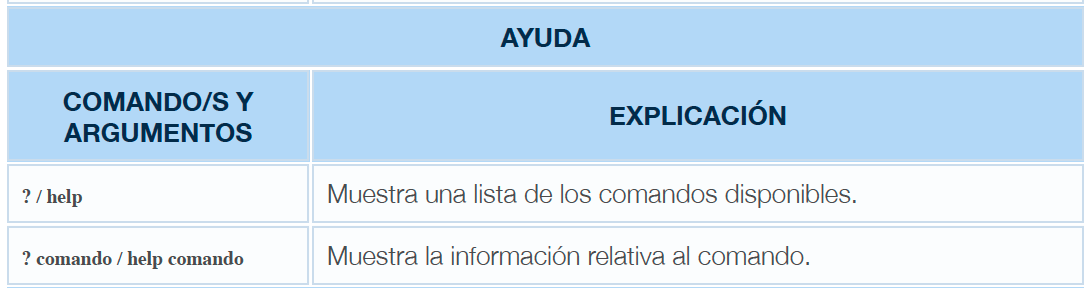
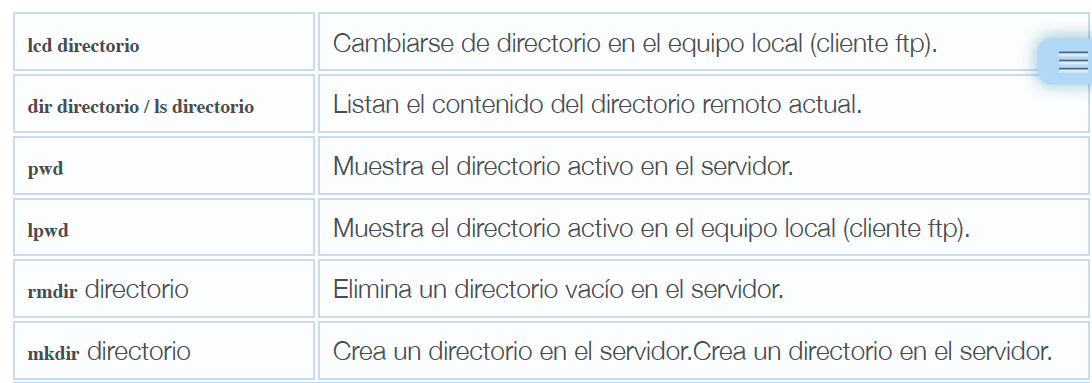
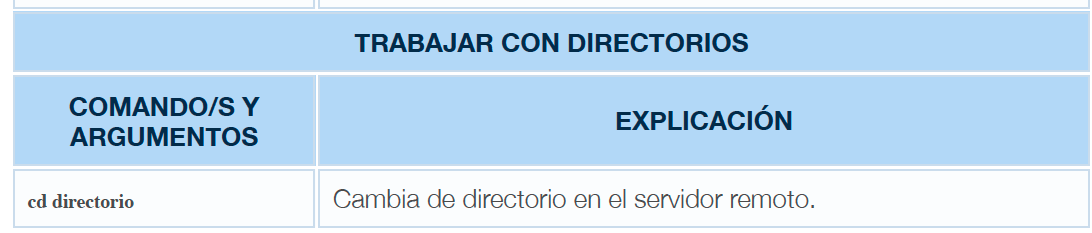
Ftp ftp.rediris.es

3. Usuario y contraseña para establecer la conexión.

4. Ahora en la consola ftp puedes ejecutar comandos. help



## 1.7.1.- Comandos ftp.



* Prompt (quitar verificación)
* !dir
* !mkdir
* Hash (almudilla=barra de estado)
* Status

## 1.8.- Servicio de transferencia de archivos en modo gráfico.

Para trabajar de un modo más interactivo se utiliza clientes ftp de modo gráfico o mediante el navegador.

Los clientes gráficos se comportan todos igual, esto es, tienen una interfaz parecida, básicamente presentan una ventana partida en dos secciones:

* la de la izquierda suele representar el equipo cliente ftp -*desde donde se intenta establecer la conexión*- y la de la derecha suele representar el equipo servidor ftp -*quién recibe la conexión*-. Luego suelen existir,en alguna zona determinada de la ventana: en el centro entre las dos secciones, arriba de las dos secciones, etc una serie de botones, usualmente representados como flechas que indican la posibilidad de subir o descargar archivos.

Dentro de los clientes ftp en modo gráfico cabe destacar dos: **gftp** y **filezilla**.

1. Cliente en modo gráfico gftp: Servidor, puerto, usuario, contraseña
2. Cliente en modo gráfico filezilla: servidor, nombre de usuario, contraseña, puerto

Para instalar el cliente gráfico FileZilla, seguiremos los siguientes pasos:

* Actualizar sistema operativo: apt get update
* Instalar cliente FTP en modo gráfico FileZilla: apt install filezilla

Forma de conexión en **mac**:

* sftp daw@192.168.56.56
* file://192.168.56.56 (user y pass autenticados del servidor) (utilizando el finder)

## 1.9.- Servicio de transferencia de archivos desde el navegador.

El navegador web es una de las herramientas más usadas para transferencia de archivos y se encuentra en la mayoría de sistemas operativos.

En el navegador web: [**ftp://nombre\_servidor\_ftp:puerto**](ftp://nombre_servidor_ftp:puerto)

Si el servidor ftp permite la conexión a un usuario anónimo entraremos directamente.

Para descargar las carpetas o archivos simplemente debes pulsar en Guardar enlace como, con el botón derecho.

Con el navegador web no soporta ni reanuda la transferencia de archivos de gran tamaño (>4GB).

## 1.10.- Asegurando el servicio de transferencia de archivos.

El protocolo ftp no cifra los datos en la transferencia de archivos, **estos viajan en texto claro.**

Cuando interese asegurar el servicio de transferencia debemos pensar en otras alternativas como: **ftps o sftp.**

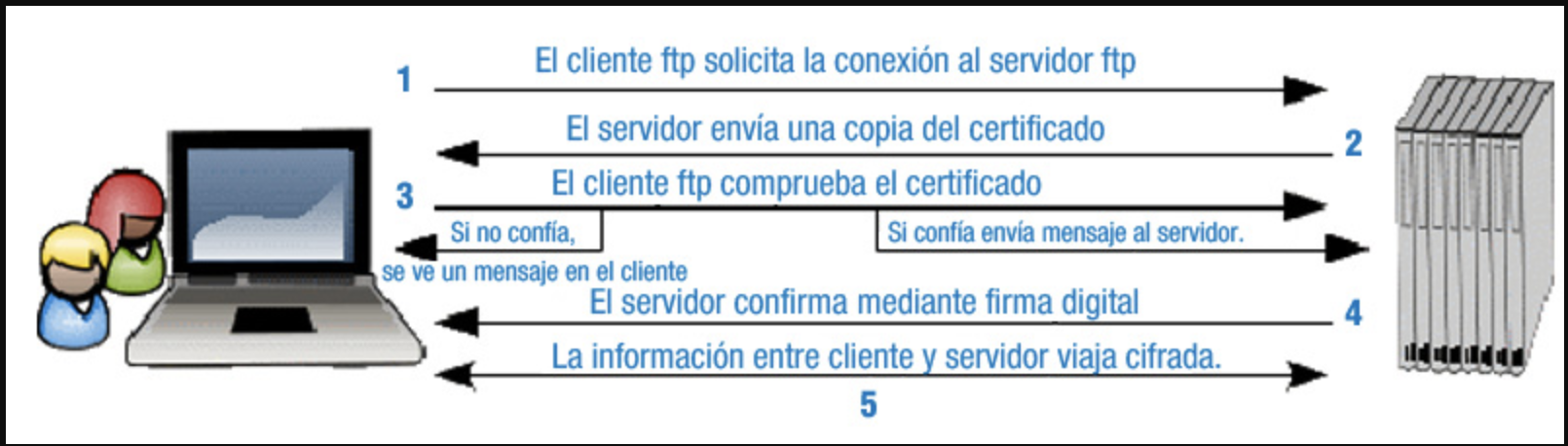
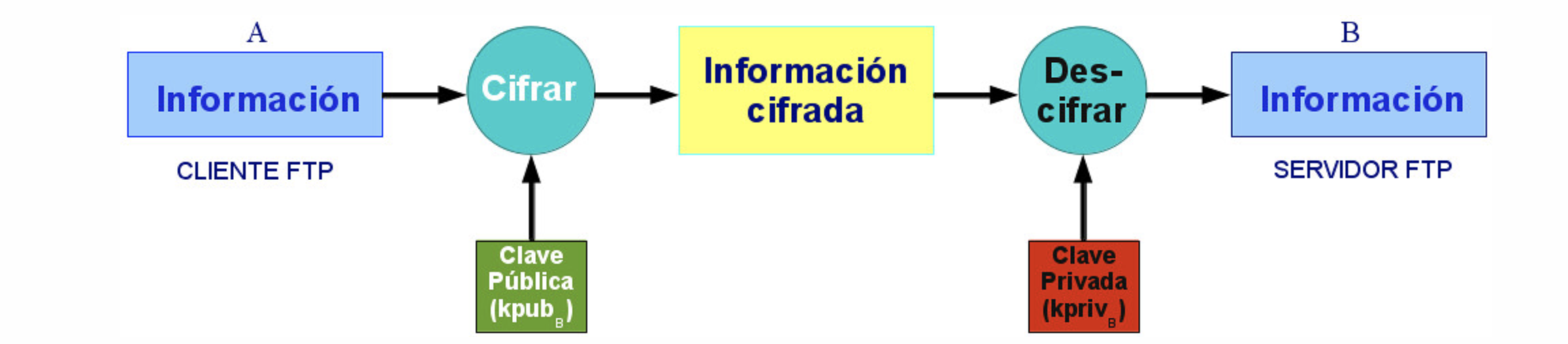
FTPS es una extensión del protocolo FTP que asegura el cifrado en la transferencia mediante los

protocolos SSL/TLS. Permite **tres** tipos de funcionamiento:

* SSL Implícito:
  + Como conexiones HTTPS
  + Usa los puertos 990 y 989.
* SSL Explícito
  + El cliente usa los mismos puertos estándar FTP: 20 y 21 pero se efectúa el cifrado en ellos.
  + Usa AUTH SSL.
* TLS Explícito:
  + Similar a SSL Explícito pero usa AUTH TLS.

El cifrado al que nos referimos es el **cifrado de clave pública o asimétrico:** clave pública (kpub) y clave privada (kpriv). La kpub interesa publicarla para que llegue a ser conocida por cualquiera, la kpriv no interesa que nadie la posea, solo el propietario de la misma. Ambas son necesarias para que la comunicación sea posible, una sin la otra no tienen sentido, así una información cifrada mediante la kpub solamente puede ser descifrada mediante la kpriv y una información cifrada mediante la kpriv sólo puede ser descifrada mediante la kpub.

En el cifrado asimétrico podemos estar hablando de individuos o de máquinas, en nuestro caso hablamos de máquinas y de flujo de información entre el cliente ftp (A) y el servidor ftp (B).



## 1.11.- El servicio de transferencia de archivos en el proceso de despliegue de una aplicación web.

Se suele configurar el servidor web con unos parámetros: tiempo de conexión y tamaño máximo de subidas de archivos diferentes del servidor ftp, de tal forma que para archivos de tamaño no muy grandes se puedan emplear aplicaciones web y no se sufra un corte en la subida de archivos y para archivos grandes, se emplee el servidor ftp.

Normalmente las empresas de alojamiento web (**hosting**) permiten la subida de archivos mediante un servidor ftp y poseen documentación para esto.

También debes saber que muchos editores web permiten subir tu aplicación web al servidor con el protocolo FTP, esto te puede resultar más sencillo que el uso de una aplicación de FTP independiente.

**Errores**:

* **Access denied**: comprobar que el usuario FTP tiene los permisos adecuados.
* Tus páginas no son reconocidas de forma automática al acceder a tu dominio: Los servidores GNU/Linux son sensibles a mayúsculas y minúsculas por lo que verifica el nombre de tus archivos.
* El cliente de FTP te muestra el mensaje **too many connections from your IP address**: esto quiere decir que existen más conexiones abiertas con el servidor FTP desde la misma dirección IP de las permitidas.

## 2.- Instalación y configuración del servidor. ProFTPD.

¿Por qué ProFTPD? Pues porque es un servidor FTP bajo licencia GPL altamente configurable, así permite:

* Usuarios virtuales con:
  + LDAP
  + BBDD: MySQL, PostgreSQL...
  + Ficheros de autenticación (ficheros de texto).
* Personalizar opciones según usuario/grupo.
* Seguridad mediante cifrado SSL/TLS.
* Configuraciones independientes mediante virtualhosts.

**Instalación y configuración:**

<https://www.redeszone.net/noticias/wifi/problemas-wifi-repetidor-evitar/>

ftpwho, ftptop

proftpd-gencert

openssl genrsa -out /etc/ssl/private/proftpd.key 4096

openssl req -new -x509 -days 1460 -key /etc/ssl/private/proftpd.key -out /etc/ssl/certs/proftpd.crt

## 2.1.- Instalación de ProFTPD

* sudo apt update
* sudo apt install proftpd
* systemctl start proftpd
* systemctl restart proftpd
* systemctl stop proftpd
* systemctl enable proftpd (para que se inicie automáticamente)

Se crean los usuarios y grupo: **proftpd y ftp, nogroup**

## 2.1.- Configuración de ProFTPD

* Fichero de configuración principal: **/etc/proftpd/proftpd.conf**
* Configurar virtual hosts, de tal forma que un mismo servidor ftp puede alojar múltiples dominios con sus configuraciones correspondientes.
* Configuración a través de directivas.
* Contexto de configuración: global, directorio, virtualhost, anonymous.
* Modularización: activar / desactivar funcionalidades a través de módulos.

Instalado ProFTPD existirán dos ficheros de interés:

* **/etc/ftpusers**: contiene una lista de usuarios que no tienen permiso de acceso por FTP. Y deben estar listados: root, bin, uucp, news.
* **/etc/proftpd/proftpd.conf:** fichero de configuración principal.
  + #, las líneas que comiencen # serán ignoradas.
  + Las líneas que comienzan por **Include** recogerán la configuración de los ficheros que la acompañan.
  + **User proftpd** y **Group nogroup**: identifican al usuario y grupo con el que se ejecuta proftpd.
  + El soporte LDAP, SQL, TLS, virtualhosts y cuotas están desactivadas.
  + El mensaje de bienvenida se encuentra en el fichero: welcome.msg
  + Conexión ftp **activo** por **defecto**.
  + Los usuarios que puedan conectarse por ftp:
    - Necesitan una consola y se puede ver en **/etc/shells**.
    - Pueden moverse por todo el sistema de ficheros: **DefaultRoot** ~ (esta comentada)
  + Para evitar ataques de denegación de servicio solamente se permite 30 conexiones simultáneas: **MaxInstances 30**
  + Permisos de ficheros y directorios: 644 y 755, **umask** 022 022
  + Al final configuración para usuarios anónimos.

Después de cualquier cambio, recargar o reiniciar el servidor.

**ProFTPD** ofrece una configuración flexible a través de varios contextos.

* Contexto: <VirtualHost 1.2.3.4> , <VirtualHost [ftp.mydomain.com](ftp://ftp.mydomain.com/)>
* Contexto: <Global>: cualquier configuración dentro de una sección <Global> es aplicada a todos los servidores del fichero incluido el servidor base también llamado **server config** server.
* Contexto: **server config** o servidor base. Esta configuración se usa para poner en marcha un servidor FTP que se pone a la escucha en todas las direcciones IP de la máquina. Se trata de la configuración del servidor que se activa como funcionalidad básica del paquete ProFTPD.

## 2.2.- Configurar el servidor como ftp privado.

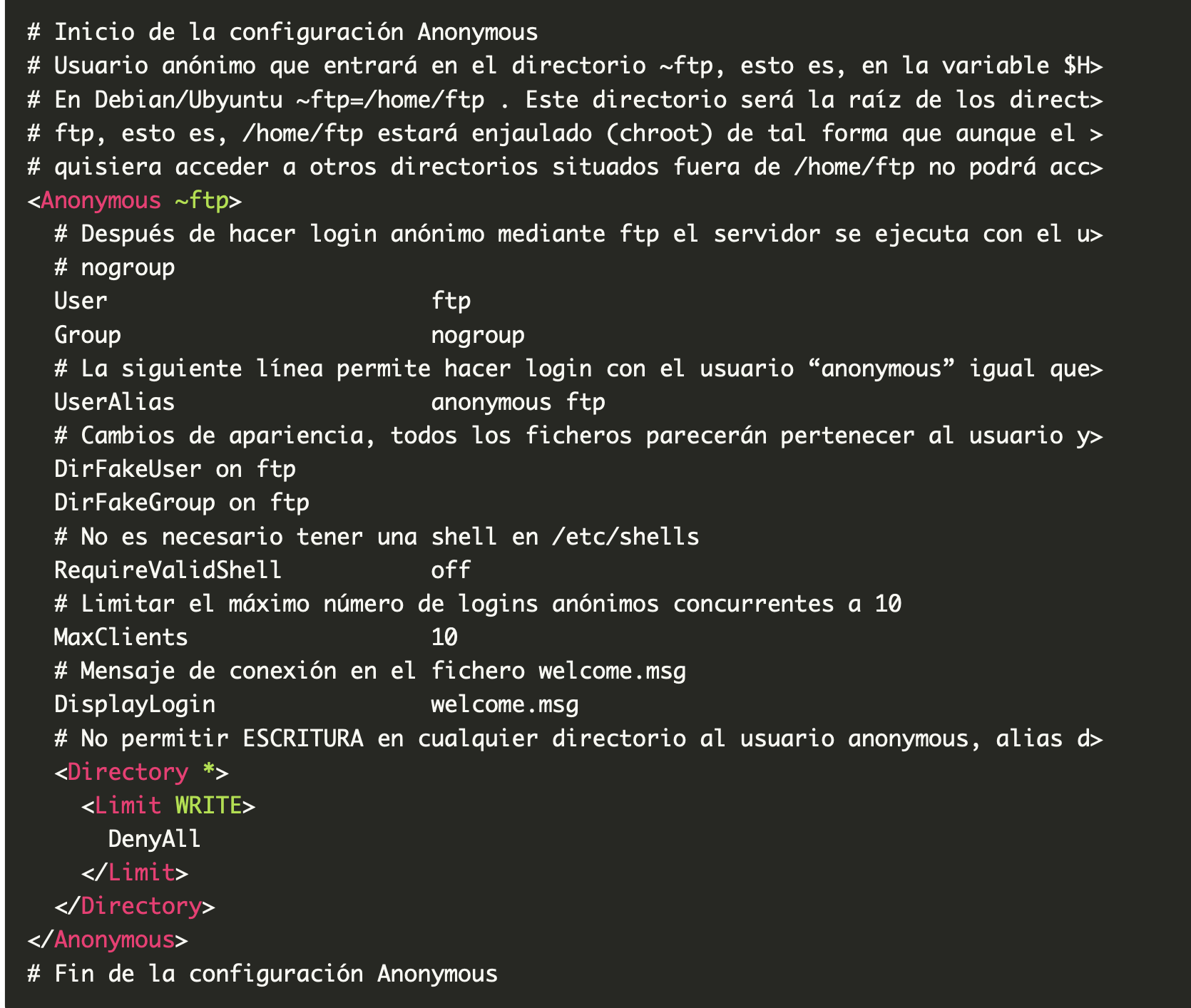
Una vez instalado el servidor ProFTPD, en Ubuntu, disponemos de un archivo de configuración **/etc/proftpd/proftpd.conf.**

Como has podido comprobar en la sección anterior, posee una **configuración tipo por defecto**. Ésta ya permite la conexión a tu servidor. ¿Con qué usuarios? Con cualquier usuario del sistema que posea una consola de comandos activa definida en **/etc/shells.** Y con cualquier cliente ftp.

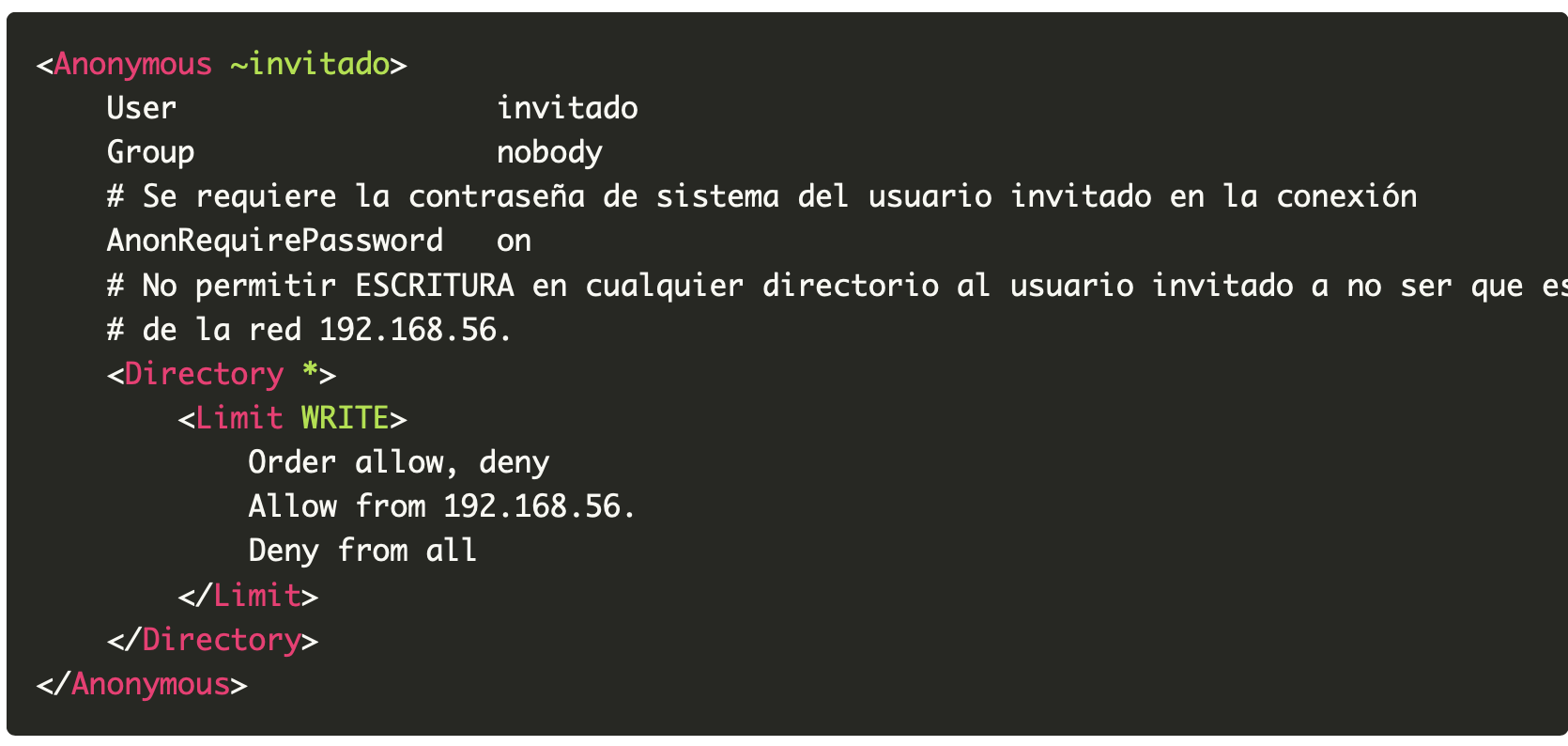
## 2.3.- Configurar el servidor como ftp privado y anónimo.

Igual te interesa contar en la configuración de tu servidor ftp con un usuario anónimo, el cual establecerá la conexión con cualquier contraseña y tendrá permisos diferentes a los usuarios del sistema (privados). Sólo podrá moverse por los directorios y descargar archivos, No podrá ni crear ni eliminar ficheros y ni directorios.

Para hacer que **proftpd** permita conexiones mediante usuarios del sistema y usuarios anónimos debes modificar el fichero **/etc/proftpd/proftpd.conf.**



Crear un usuario invitado: adduser invitado (solicitará una contraseña)



## 2.4.- Configurar el servidor ftp con usuarios virtuales.

Además de los usuarios creados en el sistema operativo donde se ha instalado el servidor ProFTPD, se pueden crear usuarios virtuales que solo existen en la configuración del servicio FTP y que **no requieren** de cuenta en el sistema operativo.

En el servidor ProFTPD se pueden crear los usuarios virtuales que quieras en un fichero similar a **/etc/passwd** y llamarlo en la configuración mediante la directiva **AuthUserFile**, entonces:

$ sudo mkdir /srv/ftp/dawdftp1 /srv/ftp/dawdftp2

$ sudo chown ftp:nogroup /srv/ftp/dawdftp1 /srv/ftp/dawdftp2

Ejecuta el siguiente comando que creará un fichero de autenticación para usuarios virtuales

ftpasswd --passwd --name dawdftp1 --file /etc/passwd.usuarios.virtuales --uid <uid-ftp> --home /srv/ftp/dawdftp1 --shell /bin/false

ftpasswd --passwd --name dawdftp2 --file /etc/passwd.usuarios.virtuales --uid <uid-ftp> --home /srv/ftp/dawdftp2 --shell /bin/false

Añadir al fichero de configuración, **proftpd.conf,** la siguiente directiva para que se puedan validar las conexiones de dichos usuarios:

AuthUserFile /etc/passwd.usuarios.virtuales

Prueba con el servidor.

## 2.5.- Virtualhosts basados en nombre

Procedimiento:

* Añadir los nombres de los equipos al archivo **/etc/hosts** para que resuelva los nombres de dominios en el servidor.

192.168.56.56   ftp.dawdistanciap2121.net

192.168.56.56   ftp.dawdistancia.net

10.0.2.2        ftp.dawdistancia2.net

* Configurar **proftpd.conf** al fichero **virtuals.conf** (descomentando la línea).
* Agregar la configuración: **/etc/proftpd/virtuals.conf**



* Recargar la configuración.

**Explicación fichero virtualhost:**

**<VirtualHost Nombre\_DNS\_servidor\_FTP>** → define el nombre de dominio del servidor ftp. También puede ser <VirtualHost Dir\_IP\_Servidor\_FTP>

**Port** numero → Identifica el puerto TCP por el que espera la conexión el servidor FTP

**ServerName** "Servidor FTP empresaX"→ Configura el nombre que se muestra en la conexión de los usuarios.

**AuthUserFile** Ruta\_Fichero → Configura los usuarios dados de alta en el servidor FTP .

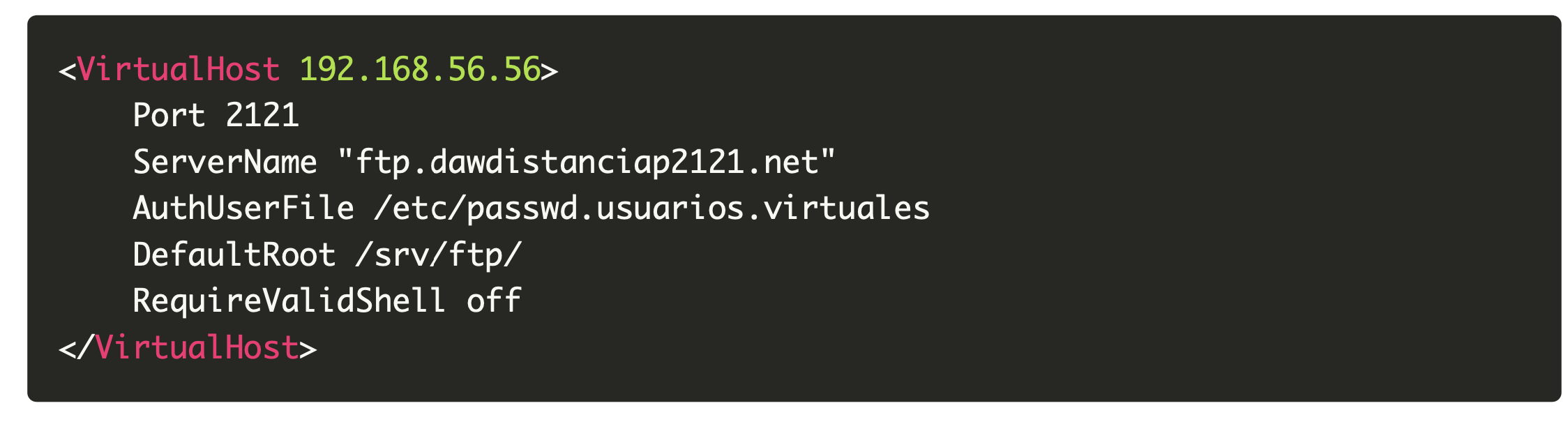
**DefaultRoot** /srv/ftp/ → Definición de la ruta que sirve ProFTPD, en este caso: /srv/ftp/ mediante la directiva DefaultRoot, esto es, indica que los usuarios cuando conecten con el servidor ftp estarán **enjaulados** en la ruta **/srv/ftp/**, con lo cual no podrán acceder a otro directorio que no esté contenido dentro de éste.

**RequireValidShell** off → No es necesario tener una shell declarada en el fichero **/etc/shells**

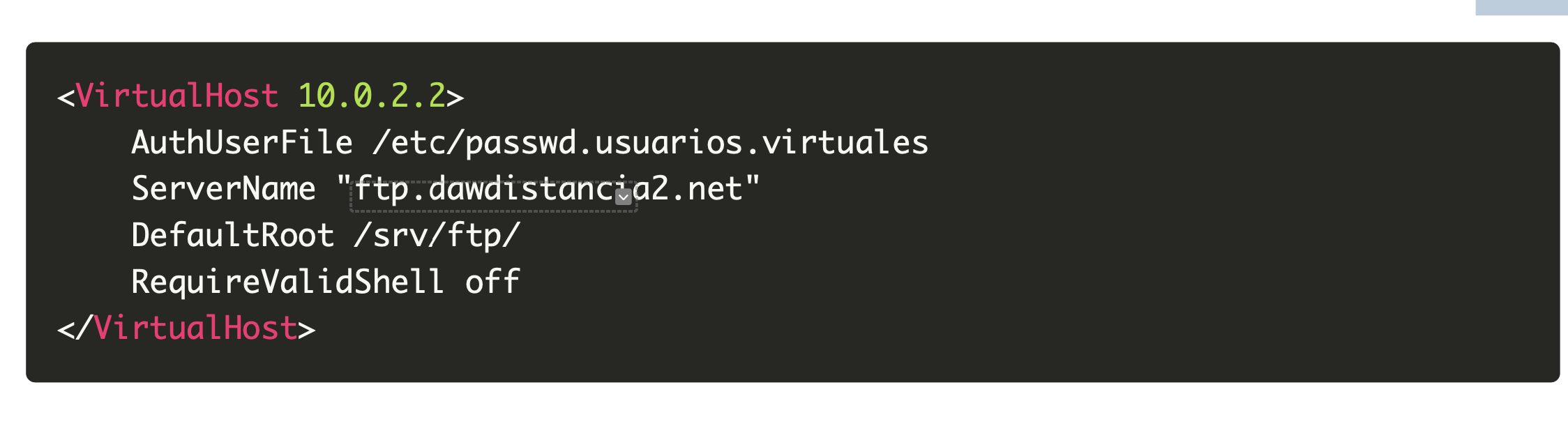
</**VirtualHost**> → fin de la definición de este virtualhost para la empresa1.

## 2.6.- Virtualhosts basados en IP.

* Descomentar la línea **virtuals.conf** en el fichero **proftpd.conf.**
* Añadir la siguiente línea en el fichero proftpd.conf (distintas IPs del servidor físico):
  + SocketBindTight on
* Añadir la configuración al fichero **virtuals.conf**



* Agrega la configuración VirtualHost para el servidor que recibe peticiones en el puerto estándar (21) de la dirección IP 10.0.2.2 en el fichero /etc/proftpd/virtuals.conf



* Recargar la configuración.

**Explicación de la sección <VirtualHost>:**

**<VirtualHost IP\_Servidor\_FTP>** → Inicio etiqueta virtualhost: define la IP del servidor ftp. También puede ser <VirtualHost Nombre\_DNS\_Servidor\_FTP>

**ServerName** "Servidor FTP empresaX"→ Configura el nombre que se muestra en la conexión de los usuarios.

**AuthUserFile** Ruta\_Fichero → Configura los usuarios dados de alta en el servidor FTP.

**DefaultRoot** /srv/ftp/ → Definición de la ruta que sirve ProFTPFD, en este caso: /var/ftp/empresaX/ mediante la directiva DefaultRoot, esto es, indica que los usuarios cuando conecten con el servidor ftp, estarán **enjaulados** en la ruta /var/ftp/empresaX/, con lo cual no podrán acceder a otro directorio que no esté contenido dentro de éste.

**RequireValidShell** off → No es necesario tener una shell declarada en el fichero /etc/shells .

</**VirtualHost**>→ Fin de la etiqueta VirtualHost: fin de la definición de este virtualhost para la empresa1.

## 2.7.- Cuotas de disco para los usuarios

La capacidad de almacenamiento no es infinita.

El archivo **/etc/proftpd/proftpd.conf** llama mediante la directiva **Include** al archivo **/etc/proftpd/modules.conf** en el que están activadas las cuotas (LoadModule mod\_quotatab.c, LoadModule mod\_quotatab\_file.c), luego para activarlas tienes que sustituir en el archivo **/etc/proftpd/proftpd.conf** el código:

<IfModule mod\_quotatab.c>

QuotaEngine off

</IfModule>

Por el siguiente:

<IfModule mod\_quotatab.c>

QuotaEngine on

QuotaLog /var/log/proftpd/quota.log

<IfModule mod\_quotatab\_file.c>

QuotaLimitTable file:/etc/proftpd/ftpquota.limittab

QuotaTallyTable file:/etc/proftpd/ftpquota.tallytab

</IfModule>

</IfModule>

donde,

**<IfModule mod\_quotatab.c> … </IfModule>** → Indica que si el módulo mod\_quotatab.c está cargado en el archivo /etc/proftpd/modules.conf se realizarán las directivas que contengan.

**QuotaEngine on** → Activa las cuotas.

**QuotaLog /var/log/proftpd/quota.log** → Indica el archivo de registro sobre cuotas.

**<IfModule mod\_quotatab\_file.c> … </IfModule>** → Indica que si el módulo mod\_quotatab\_file.c está cargado en el archivo /etc/proftpd/modules.conf se realizarán las directivas que contengan.

**QuotaLimitTable file:/etc/proftpd/ftpquota.limittab** → Indica el archivo sobre el límite de cuotas Limit.

**QuotaTallyTable file:/etc/proftpd/ftpquota.tallytab** → Indica el archivo sobre el límite de cuotas Tally.

Para **ProtFTPD** existen básicamente dos tipos de cuotas: **limit** y **tally**.

**Limit**: Es la cuota que te interesa si estás pensando en restringir el espacio en disco a los usuarios. Éste puede ser **soft**, cuando existe un espacio de gracia(tamaño en bytes) que puede sobrepasar el límite, o **hard** cuando no existe un espacio de gracia.

**Tally**: Utilizado cuando quieres limitar el número de ficheros que se utilizan.

La forma más sencilla de crear las cuotas es hacer el símil **upload=espacio en disco**, con lo cual los archivos subidos no pueden ocupar más del que queramos darle como espacio en disco, esto es, los bytes subidos funcionan como espacio en disco, ya que no existe diferencia entre ellos, pues los bytes cargados a través de FTP se almacenan automáticamente en el disco, por lo que deberías emplear la cuota tipo **limit**.

Puedes crear las cuotas mediante el comando **ftpquota**:

$ sudo ftpquota --create-table --type=limit --table-path=/etc/proftpd/ftpquota.limittab

$ sudo ftpquota --create-table --type=tally --table-path=/etc/proftpd/ftpquota.tallytab

Por ejemplo, si quisieras limitar a un usuario de nombre **dawdftp1** el espacio de subida en 4 GB:

$ sudo ftpquota --add-record --type=limit --name=dawdftp1 --quota-type=user --bytes-upload=4 --units=Gb --table-path=/etc/proftpd/ftpquota.limittab

Y si quisieras limitar la subida y bajada a 4 GB y 2 GB respectivamente al usuario dawdftp2:

$ sudo ftpquota --add-record --type=limit --name=dawdftp2 --quota-type=user --bytes-upload=4 --bytes-download=2 --units=Gb --table-path=/etc/proftpd/ftpquota.limittab

Pues, utilizas el comando **ftpquota** como sigue:

Para ver los registros de cuotas, esto es, a quién se le está ejerciendo las cuotas:

$ sudo ftpquota --show-records --type=limit --table-path=/etc/proftpd/ftpquota.limittab



Para actualizar la cuota de un usuario, por ejemplo, user-empresa1:

$ sudo ftpquota --update-record --type=limit --name=dawdftp1 --quota-type=user --bytes-upload=2300 --units=Mb --table-path=/etc/proftpd/ftpquota.limittab

con lo cual, si compruebas de nuevo los registros, verás que los cambios surgieron efecto:



Para desactivar la cuota de un usuario debes borrar el registro, por ejemplo, user-empresa1:

$ sudo ftpquota --delete-record --type=limit --name=dawdftp1 --quota-type=user

con lo cual, si compruebas de nuevo los registros, verás que los cambios surgieron efecto:

$ sudo ftpquota --show-records --type=limit --table-path=/etc/proftpd/ftpquota.limittab

ftpquota: (empty table)

2.8.- Acceso seguro mediante TLS.

Primero habrá que instalar el **módulo tls** que nos va a proporcionar la capa de seguridad sobre la que van a enviarse las comunicaciones del protocolo FTP:

$ sudo apt install proftpd-mod-crypto

Después debemos habilitar el módulo tls, **descomentando** la siguiente línea del fichero **modules.conf** en el directorio **/etc/proftpd**:

LoadModule mod\_tls.c

Edita el archivo **/etc/proftpd/proftpd.conf** y descomenta la línea:

Include /etc/proftpd/tls.conf

Crea las claves, pública y privada, para la conexión cifrada:

Método 1: Instalación del paquete y ejecución del comando openssl.



Método 2: Comando **proftpd-gencert**. Los dos comandos anteriores se resumen en uno, con la salvedad que el certificado será válido solamente durante 1 año y no 10:

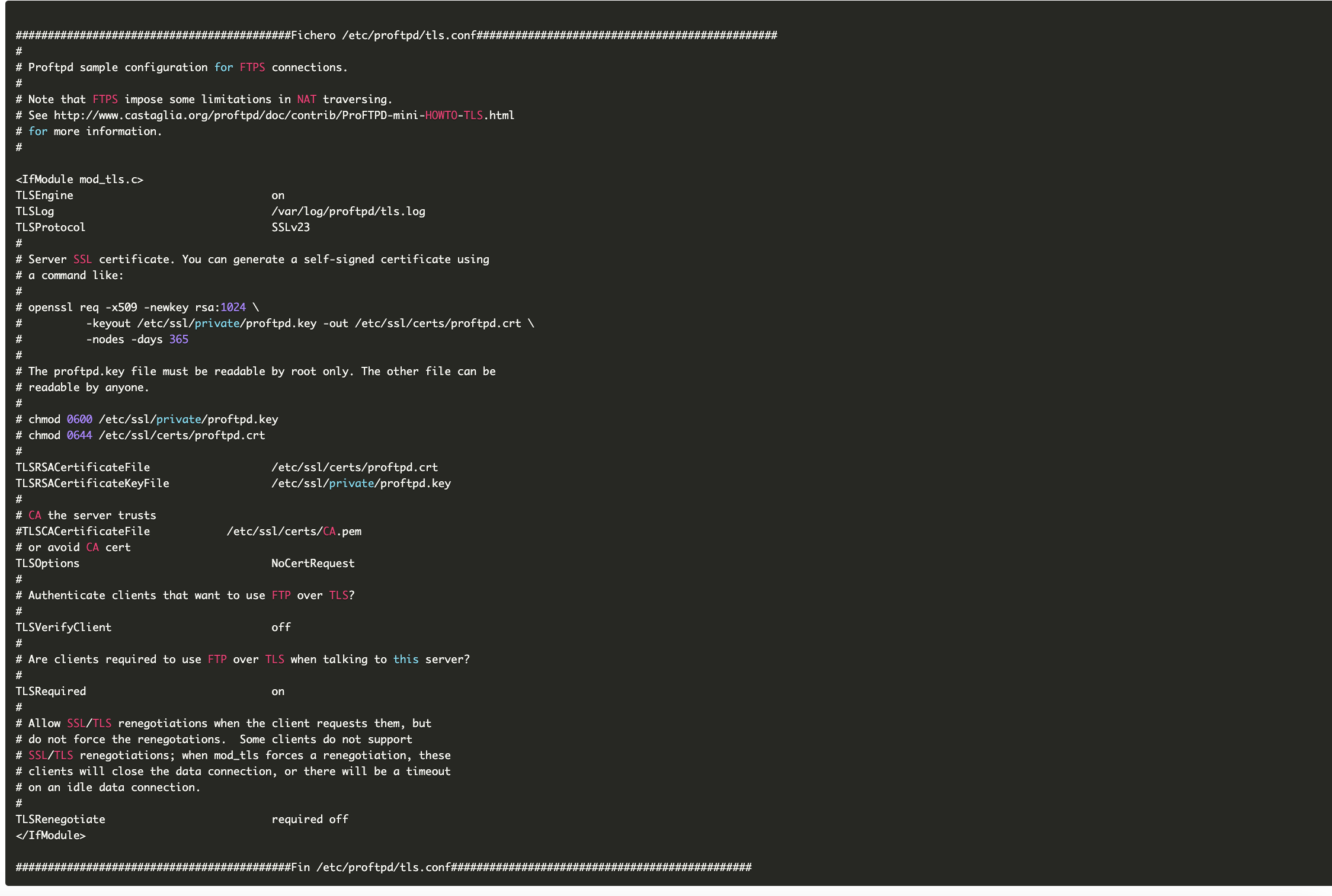
$ proftpd-gencert

Modifica los permisos:

$ sudo chmod 0600 /etc/ssl/private/proftpd.key

$ sudo chmod 0644 /etc/ssl/certs/proftpd.crt

Modifica el fichero **/etc/proftpd/tls.conf** para que tenga el siguiente contenido:



Recarga la configuración del servidor ProFTPD:

$ sudo systemctl restart proftpd

Comprueba la configuración con un cliente FTPS, es decir, un cliente ftp que permita la conexión por TLS como FileZilla.

Puedes verificar en tiempo real las conexiones con el servidor ftp revisando los archivos de registro mediante los comandos:

$ tail -f /var/log/proftpd/proftpd.log

$ tail -f /var/log/proftpd/tls.log

Si deseas, puedes hacer valer la configuración para todos los usuarios, incluso aquellos pertenecientes a virtualhost, modificando el fichero **/etc/proftpd/tls.conf** como se indica en el fichero ejemplo [tls2.conf](https://aulavirtual32.educa.madrid.org/ies.lapaz.alcobendas/pluginfile.php/66319/mod_resource/content/18/DAW04_proftpd_tls2.conf).

## 2.9.- Creación de políticas de autorización

La etiqueta **<Limit>** en **ProFTPD** es una herramienta esencial para controlar el acceso y las acciones que los usuarios pueden realizar en tu servidor FTP. Se utiliza para aplicar restricciones sobre una amplia gama de operaciones FTP, permitiendo a los administradores del servidor definir políticas de seguridad detalladas y adaptadas a sus necesidades específicas. Dentro de un bloque **<Limit>**, puedes especificar varias operaciones FTP para restringir, tales como:

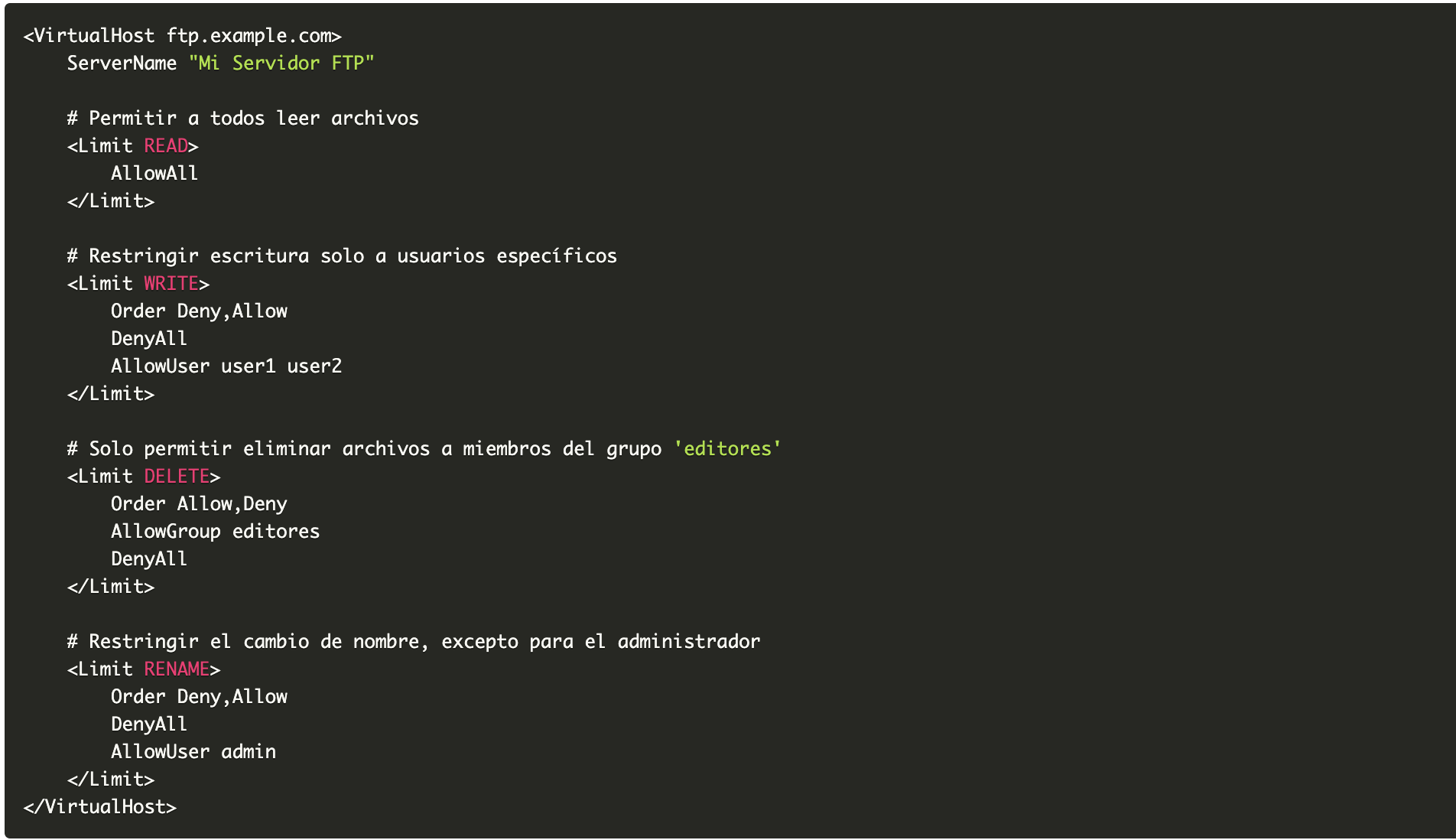
* **READ**: Limita operaciones de lectura como RETR (descargar archivos).
* **WRITE**: Restringe operaciones de escritura como STOR (subir archivos), STOU, APPE.
* **DIRS**: Afecta comandos que listan directorios o cambian directorios, como LIST, NLST, CWD, CDUP.
* **DELETE**: Restringe la eliminación de archivos y directorios con DELE, RMD.
* **RENAME**: Controla el cambio de nombre de archivos y directorios con RNFR, RNTO.
* **ALL**: Aplica restricciones a todas las operaciones FTP disponibles.

Uso de Directivas **Allow**, **Deny** y **Order**Las directivas **Allow** y **Deny** se utilizan dentro de un bloque **<Limit>** para especificar quién tiene permitido o denegado realizar las operaciones restringidas. La directiva **Order** determina el orden en que se evalúan estas reglas.

**Ejemplo Completo**  
Supongamos que quieres configurar un servidor FTP con las siguientes políticas:

* Todos los usuarios pueden leer archivos (READ), pero solo ciertos usuarios pueden subir o modificar archivos (WRITE).
* Solo un grupo específico puede eliminar archivos (DELETE).
* Nadie puede renombrar archivos (RENAME), excepto el administrador.

La configuración en proftpd.conf podría verse así:



* **<Limit READ>**: Esta sección permite a todos los usuarios realizar operaciones de lectura. No se necesita Order porque AllowAll es la única regla.
* **<Limit WRITE>**: Aquí se utiliza Order Deny,Allow para primero bloquear a todos (DenyAll) y luego permitir explícitamente a user1 y user2. Esto significa que, por defecto, nadie puede escribir, excepto estos dos usuarios.
* **<Limit DELETE>**: Se configura con Order Allow,Deny, lo que significa que primero se procesan las directivas Allow. En este caso, los miembros del grupo editores pueden eliminar archivos, y luego DenyAll bloquea esta acción a cualquier otro usuario que no sea miembro del grupo.
* **<Limit RENAME>**: Similar a WRITE, se niega la acción a todos excepto al usuario admin, que tiene permiso explícito para renombrar archivos.

La directiva **Order** es crucial porque define cómo se aplican las reglas **Allow** y **Deny**. **Allow,Deny** significa que si un usuario pasa una regla **Allow**, se le permitirá la acción a menos que una regla **Deny** también se aplique. **Deny,Allow** significa que se aplican todas las restricciones hasta que una regla Allow específicamente concede acceso.  
Estas configuraciones permiten crear un entorno FTP más seguro y controlado, donde solo usuarios y grupos específicos tienen permisos para realizar ciertas acciones, limitando la posibilidad de acciones no autorizadas.

Fin.