

2º DESARROLLO DE APLICACIONES WEB  
29/11/2020

NEREA ÁLVAREZ JUSTEL

# Protocolo de transferencia de archivos

Despliegue de aplicaciones web (DAW)

## Contenido

<b>INTRODUCCIÓN .....</b>	<b>2</b>
<b>DESARROLLO DE LOS CONTENIDOS .....</b>	<b>3</b>
1. Explica y enumera protocolos asociados a la transferencia de archivos a un servidor remoto: TFTP, FTP, FTPS, FTPES, SFTP, FXP.....	3
2. Compara entre los protocolos SFTP vs FTP. ....	6
3. Comparación entre FTP implícito y explícito.....	9
4. Se pide la creación de un script en Linux (altaUsuarios.bash) para la creación de usuarios enjaulados en un servidor web. ....	10
5. Se pide la creación de un script en Linux(bajaUsuarios.sh) para la eliminación de usuarios anteriores.....	11
6. Paso a seguir o lo que hay que tener en cuenta para que dichos usuarios estén enjaulados.....	12
7. Control de acceso al servicio SSH, solo pueden acceder los usuarios que pertenecen al grupo ftpusers y el usuario miadmin. ....	16
8. Investiga como controlar la cuota de disco asignada a cada usuario Linux. ....	17
<b>CONCLUSIÓN .....</b>	<b>18</b>
<b>BIBLIOGRAFÍA.....</b>	<b>19</b>

## INTRODUCCIÓN

Aprenderemos Información sobre los protocolos y sus tipos al igual que la creación de usuarios y su enjaulado, permisos y acceso.

## DESARROLLO DE LOS CONTENIDOS

### 1. Explica y enumera protocolos asociados a la transferencia de archivos a un servidor remoto: TFTP, FTP, FTPS, FTPES, SFTP, FXP.

#### ¿Qué es FTP?

FTP (File Transfer Protocol) es un protocolo de transferencia de archivos cliente-servidor creado en 1971. Su funcionamiento es el siguiente: un PC cliente inicia una conexión a un servidor y luego le envía archivos o descarga archivos del servidor. La conexión cliente->servidor se llama "modo pasivo" mientras que la conexión servidor a cliente es el "modo activo".

Ya en los años '90, se sabía que el tráfico FTP entre un PC cliente y una máquina servidor era inseguro, y que se podían capturar fácilmente datos como el usuario y el password con el que un cliente se conectaba a un servidor. Esto es posible porque, en conexiones FTP, las credenciales son enviadas en texto plano desde el cliente hacia el servidor. Como no hay ningún tipo de encriptación, si hay alguien analizando el tráfico en la red local donde se encuentra el PC cliente o en la red donde se encuentra el servidor, esta persona podrá capturar el usuario y password de conexión al servidor, así como crear copias de los archivos enviados y recibidos.

#### ¿Qué es FTPS?

FTPS es la evolución de FTP, añadiendo cifrado por SSL/TLS en las conexiones para encriptar los datos enviados en una sesión cliente-servidor. De esta forma, usuario, password y datos enviados quedan cifrados y ya no son visibles en texto plano en caso de que alguien intercepte paquetes en nuestra red.

El cifrado de las transferencias se solía realizar a través del protocolo Secure Sockets Layer (SSL), aunque hoy en día se usa su sucesor, Transport Layer Security o TLS. Por lo tanto, cuando hablamos de FTPS estamos hablando de FTP sobre TLS.

Hay dos tipos de configuraciones FTPS:

- FTP sobre TLS (explicit encryption): el cliente establece una conexión TCP al puerto X del servidor y comienza una sesión FTP estándar, es decir, sin cifrar. En el momento en que el servidor pide al cliente las credenciales, se establece una segunda conexión sobre TLS, por la cual se envían los datos sensibles. Esta conexión se pone en marcha usando el comando AUTHSSL. Añadir que anteriormente era FTP sobre SSL, pero se cambió el protocolo SSL por TLS cuando se creó el protocolo TLS.

- FTP sobre SSL (implicit encryption): es un método más antiguo y, por ello, obsoleto en favor de las conexiones explícitas. El cliente se conecta a un puerto distinto del servidor (normalmente el 990) y se realiza una negociación SSL previa a enviar cualquier comando. Es decir, el canal seguro es establecido antes de ejecutar comandos en el servidor.

### ¿Qué es FTPES?

FTPES no es más que FTPS explícito, de ahí la E.

### ¿Qué es SFTP?

SFTP es la abreviatura de Secure File Transfer Protocol (Protocolo de transferencia segura de archivos). Este protocolo permite transferir datos cifrados entre dos máquinas cliente-servidor.

El estándar SFTP fue desarrollado por el IETF (Internet Engineering Task Force) como una extensión de la segunda versión del SSH (Secure Shell Protocol) para proporcionar a los usuarios una transferencia segura de archivos. SFTP se ha convertido con el tiempo en el estándar de oro en el campo de los protocolos de transferencia de archivos por su seguridad, facilidad de uso y versatilidad.

El IETF afirma que, aunque SFTP está definido en el contexto del protocolo SSH2, SFTP es en realidad un estándar independiente del resto del conjunto de protocolos SSH2 (por lo que no está limitado por los propios conceptos y definiciones del SSH2).

Aunque SFTP es nativo de entornos \*NIX, también se puede usar en Windows con programas de terceros o con los nuevos builds de OpenSSH para Windows que Microsoft lanzó el año pasado.

### ¿Qué es FXP?

No mucha gente conoce el protocolo FXP. FXP (File eXchange Protocol) es un método de transferencia de datos entre dos servidores orquestado por un cliente, es decir, un PC controla las transferencias entre dos servidores sin que los archivos pasen por el PC cliente.

Para ponerlo en perspectiva, la comunicación convencional FTP consiste en un solo servidor y cliente y la transferencia de datos se realiza entre ambos. Durante una sesión FXP, en cambio, un cliente mantiene conexiones estándares con dos servidores, dirigiendo cualquiera de los dos servidores que se conecte al otro para iniciar una transferencia de datos. Este método permite a un cliente con poco ancho de banda intercambiar datos entre dos servidores con más ancho de banda sin el retraso asociado a recibir él los archivos del servidor 1 para enviarlos posteriormente al servidor 2. A lo largo de este proceso, sólo el cliente es capaz de acceder a los recursos de los dos servidores.

FXP tuvo su auge a principios de los 2000, pero ha caído en desuso, ya que usar FXP expone a un servidor a sufrir un ataque de FTP bounce, lo cual supone un riesgo de seguridad para toda la red local.

### **¿Qué es TFTP?**

TFTP son las siglas de Trivial file transfer Protocolo (Protocolo de transferencia de archivos trivial). TFTP es un protocolo de transferencia de archivos muy simple, el cual no requiere usuario ni password. Se usa para transferir pequeños archivos entre dos dispositivos conectados entre sí. Por ejemplo, se puede usar para actualizar el firmware de un switch, entre otras cosas.

Características del TFTP:

- Utiliza UDP (puerto 69) como protocolo de transporte.
- No puede listar el contenido de los directorios.
- No existen mecanismos de autenticación (usuario/contraseña) o cifrado.
- Soporta tres modos diferentes de transferencia, "netascii", "octet" y "mail". Los dos primeros corresponden a los modos "ascii" e "imagen" (binario) del protocolo FTP.

## 2. Compara entre los protocolos SFTP vs FTP.

La tarea más común del entorno de red es transferir los archivos / datos / información entre los hosts en la red. **FTP y SFTP** son los **protocolos de transferencia de archivos**. La transferencia de archivos a través de la red en formato de texto sin formato puede plantear problemas de seguridad. El protocolo FTP se introdujo cuando la seguridad en Internet no era un gran problema. Los datos se enviaron sin cifrar en FTP, lo que puede ser fácilmente interceptado por el atacante. Por lo tanto, se requirió algún canal seguro para transferir los archivos. Para esto, uno puede agregar una **Capa de sockets seguros** entre la capa de aplicación FTP y TCP o simplemente puede usar un protocolo independiente llamado SFTP.

FTP y SFTP transfieren el archivo de una computadora a otra, pero la diferencia básica entre FTP y SFTP es que **FTP** no proporciona un canal seguro para transferir archivos, mientras que **SFTP** sí lo hace.

Bases para comparar	FTP	SFTP
BASIC	FTP no proporciona un canal seguro para transferir archivos entre hosts.	SFTP proporciona un canal seguro para transferir los archivos entre los hosts.
Forma completa	Protocolo de transferencia de archivos.	Protocolo seguro de transferencia de archivos.
Protocolo	FTP es un protocolo TCP / IP.	El protocolo SFTP es una parte del protocolo SSH (un programa de aplicación de inicio de sesión remoto).
Conexión	FTP establece conexión de control en el puerto TCP 21.	SFTP transfiere el archivo bajo la conexión establecida por el protocolo SSH entre el cliente y el servidor. (Puerto 22)
Cifrado	La contraseña y los datos de FTP se envían en formato de texto simple.	SFTP encripta los datos antes de enviarlos.

### Definición de FTP

FTP (**File Transfer Protocol**) es un protocolo en TCP / IP que copia un archivo de un host a otro host. Sin embargo. Hay algunos problemas, ya que los dos sistemas que envían y reciben archivos pueden tener una *forma diferente de representar los datos*; pueden tener *diferentes convenciones de nombre de archivo*, pueden tener *diferentes estructuras de directorio*.

FTP proporciona una solución simple a todos los problemas anteriores. FTP es diferente de otra aplicación cliente-servidor que establece **dos conexiones** entre los hosts que se comunican. Una conexión es para la **transferencia de datos** y otra para la **información de control** (comando y respuestas). FTP es más eficiente que otras aplicaciones cliente-servidor, ya que tiene una conexión separada para datos y comandos.

La conexión de control es simple, ya que es solo para establecer una conexión entre los hosts. Pero la conexión de datos es compleja ya que tiene que transferir la **variedad de datos**. El FTP establece la **conexión de control** en el puerto número **21 de TCP** y la **conexión de datos** en el puerto número **20 de TCP**.

### Definición de SFTP

SFTP (Secure File Transfer Protocol) es una forma segura de transferir los archivos a través de la red. Aunque contamos con el protocolo FTP para transferir los archivos de un host a otro en la red, pero la hora en que se diseñó FTP no fue un problema importante.

El protocolo FTP requiere la contraseña para establecer la conexión con el host al que se debe enviar el archivo, pero la contraseña está en el texto simple que amenaza con ser interceptado por un atacante. El atacante puede entonces hacer un mal uso de la contraseña. Los datos también se envían en texto sin formato a través de la conexión de datos, que de nuevo es insegura.

Entonces, SFTP introdujo un canal seguro para transferir los archivos a través de la red. SFTP es una parte del protocolo SSH (Secure Shell) que en realidad es un programa en Unix. El protocolo SSH establece una conexión segura entre el cliente y el servidor, y luego el programa SFTP funciona de manera similar a FTP y transfiere el archivo en el canal seguro creado por SSH. De esta manera, el archivo se puede transferir de forma segura utilizando SFTP.

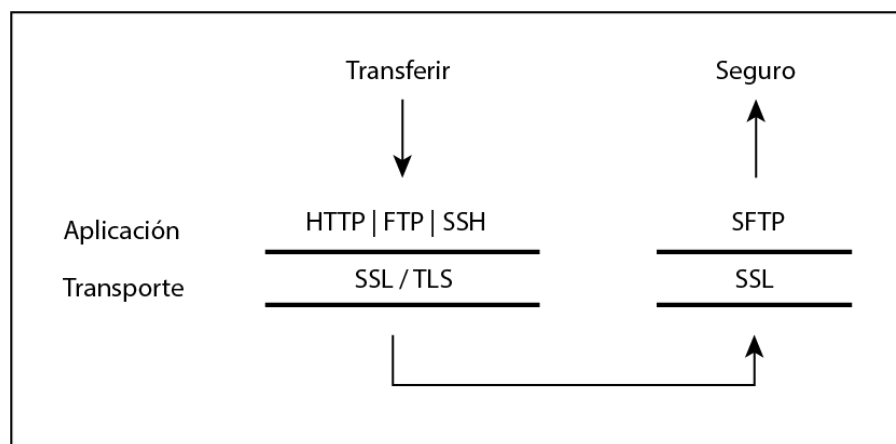


**Diferencias clave entre FTP y SFTP**

- FTP no proporciona ningún canal seguro para transferir los archivos entre los hosts, mientras que el protocolo SFTP proporciona un canal seguro para transferir los archivos entre los hosts de la red.
- FTP es una abreviatura de File Transfer Protocol, mientras que SFTP es una abreviatura de Secure File Transfer Protocol .
- El protocolo FTP es un servicio proporcionado por TCP / IP . Sin embargo, SFTP es una parte del protocolo SSH que es una información de inicio de sesión remota.
- FTP hace una conexión usando la conexión de control en el puerto TCP 21 . Por otro lado, SFTP transfiere el archivo bajo la conexión segura establecida por el protocolo SSH entre el cliente y el servidor.
- FTP transfiere la contraseña y los datos en formato de texto simple, mientras que SFTP cifra los datos antes de enviarlos a otro host.

**Conclusión:**

Tanto FTP como SFTP son el protocolo de transferencia de archivos, pero SFTP proporciona una forma segura de transferir el archivo de un host a otro host en la red.



### 3. Comparación entre FTP implícito y explícito.

Las variantes seguras de FTP incluyen FTPS Implícito SSL y FTPS Explicit SSL. Ambos utilizan cifrado SSL.

#### **SSL IMPLÍCITO DE FTPS**

En el modo SSL implícito, se establece una sesión SSL requerida entre el cliente y el servidor antes de intercambiar los datos. Como su nombre lo sugiere, el uso de SSL está implícito y cualquier intento de conexión realizado por un cliente sin usar SSL es rechazado por el servidor. Los servicios SSL implícitos de FTPS generalmente se ejecutan en el puerto 990. Aunque todavía se usan en la actualidad, muchos consideran que el SSL implícito de FTPS está obsoleto a favor del SSL explícito de FTPS.

#### **FTPS EXPLICIT SSL**

En el modo SSL explícito, el cliente y el servidor negocian el nivel de protección utilizado. Esto es muy útil porque el servidor puede admitir sesiones FTP sin cifrar y sesiones FTPS cifradas en un solo puerto. En una sesión SSL explícita, el cliente primero establece una conexión no cifrada al servicio FTP. Antes de enviar las credenciales de usuario, el cliente solicita que el servidor cambie el canal de comando a un canal cifrado SSL mediante el envío del comando AUTH TLS o AUTH SSL. Luego de la configuración exitosa del canal SSL, el cliente envía las credenciales de usuario al servidor FTP. Estas credenciales, junto con cualquier otro comando enviado al servidor durante la sesión de FTP, se encriptan automáticamente por el canal SSL. De forma similar a la forma en que se puede proteger el canal de comandos, el nivel de protección utilizado en el canal de datos se negocia entre el cliente y el servidor mediante el comando PROT.

#### 4. Se pide la creación de un script en Linux (altaUsuarios.bash) para la creación de usuarios enjaulados en un servidor web.

La creación de usuarios mediante script es tan fácil como ejecutar ese script. En nuestro caso los script son aportados por la profesora aunque puede que tengamos que cambiar algo, como en mi caso **GRUPO\_SFTP="ftpusers" y DOMINIO="nereaa.local"**.

En este enlace se puede ver los archivos con sus líneas.

<https://github.com/IESSAUCES/SERVERDAW/blob/ramaNereaA/users/crearUsuario.bash>

El fichero se ejecuta mediante el comando:

***sudo bash crearUsuario.bash***

Al ejecutarlo saldrá un listado de los usuarios según se crean, pero también podemos visualizar los usuarios creados mediante el comando.

```
miadmin@NAJUSED:~/SERVERDAW/users$ sudo bash crearUsuario.bash
CURSO 2
CURSO: 2
VALOR INICIAL 201
201
DAW201
Usuario creado correctamente DAW201
mode of '/var/www/DAW201/public_html' changed from 0755 (rwxr-xr-x) to 2775 (rwxrwsr-x)
202
DAW202
Usuario creado correctamente DAW202
mode of '/var/www/DAW202/public_html' changed from 0755 (rwxr-xr-x) to 2775 (rwxrwsr-x)
203
DAW203
Usuario creado correctamente DAW203
mode of '/var/www/DAW203/public_html' changed from 0755 (rwxr-xr-x) to 2775 (rwxrwsr-x)
204
DAW204
Usuario creado correctamente DAW204
mode of '/var/www/DAW204/public_html' changed from 0755 (rwxr-xr-x) to 2775 (rwxrwsr-x)
205
DAW205
Usuario creado correctamente DAW205
mode of '/var/www/DAW205/public_html' changed from 0755 (rwxr-xr-x) to 2775 (rwxrwsr-x)
206
DAW206
Usuario creado correctamente DAW206
```

***cat /etc/group | grep ftp***

```
miadmin@NAJUSED:~/SERVERDAW/users$ cat /etc/group | grep ftp
ftpusers:x:1001:DAW12,daw201,daw202,DAW201,DAW202,DAW203,DAW204,DAW205,DAW206,DAW207,DAW208,
DAW209,DAW210,DAW211,DAW212,DAW213,DAW214,DAW215,DAW216,DAW217
miadmin@NAJUSED:~/SERVERDAW/users$
```

## 5. Se pide la creación de un script en Linux(bajaUsuarios.sh) para la eliminación de usuarios anteriores.

La creación de usuarios mediante script es tan fácil como ejecutar ese script. En nuestro caso los script son aportados por la profesora aunque puede que tengamos que cambiar algo, como en mi caso GRUPO\_SFTP="ftpusers" y DOMINIO="nereaa.local".

En este enlace se puede ver los archivos con sus líneas.

<https://github.com/IESSAUCES/SERVERDAW/blob/ramaNereaA/users/borrarUsuario.bash>

El fichero se ejecuta mediante el comando:

***sudo bash borrarUsuario.bash***

Al ejecutarlo saldrá un listado de los usuarios según se borran, , pero tambien podemos visualizar los usuarios creados mediante el comando

```
miadmin@NAJUSED:~/SERVERDAW/users$ sudo bash borrarUsuario.bash
CURSO 2
CURSO: 2
VALOR INICIAL 201
DAW201
Usuario eliminado correctamente
Home del usuario eliminado correctamente
Usuario DAW201 eliminado correctamente!
DAW202
Usuario eliminado correctamente
Home del usuario eliminado correctamente
Usuario DAW202 eliminado correctamente!
DAW203
7 Usuario eliminado correctamente
Home del usuario eliminado correctamente
Usuario DAW203 eliminado correctamente!
DAW204
Usuario eliminado correctamente
Home del usuario eliminado correctamente
Usuario DAW204 eliminado correctamente!
DAW205
Usuario eliminado correctamente
Home del usuario eliminado correctamente
```

***cat /etc/gropu | grep ftp***

```
miadmin@NAJUSED:~/SERVERDAW/users$ cat /etc/group | grep ftp
ftpusers:x:1001:DAW12,daw201,daw202
miadmin@NAJUSED:~/SERVERDAW/users$
```

## 6. Paso a seguir o lo que hay que tener en cuenta para que dichos usuarios estén enjaulados.

Primero veremos la creación de usuario individual:

- Creación de usuario

Creación de grupo

**Sudo groupadd ftpusers**

Creación del usuario

**Sudo useradd -g www-data -G ftpusers -d /var/www/daw201 daw201**

Cambiar la contraseña

**Sudo passwd daw201**

- Creación de carpeta home

El home del usuario pertenece a root

***sudo chown root:root /var/www/daw201***

Eliminar el permiso de escritura

***sudo chmod -w /var/www/daw201***

Como el usuario no tendrá privilegios de escritura crearemos la carpeta public\_html para su uso, siendo esta de su propiedad

- Creación de la carpeta public\_html

***sudo mkdir /var/www/daw201/public\_html***

Permisos de public\_html

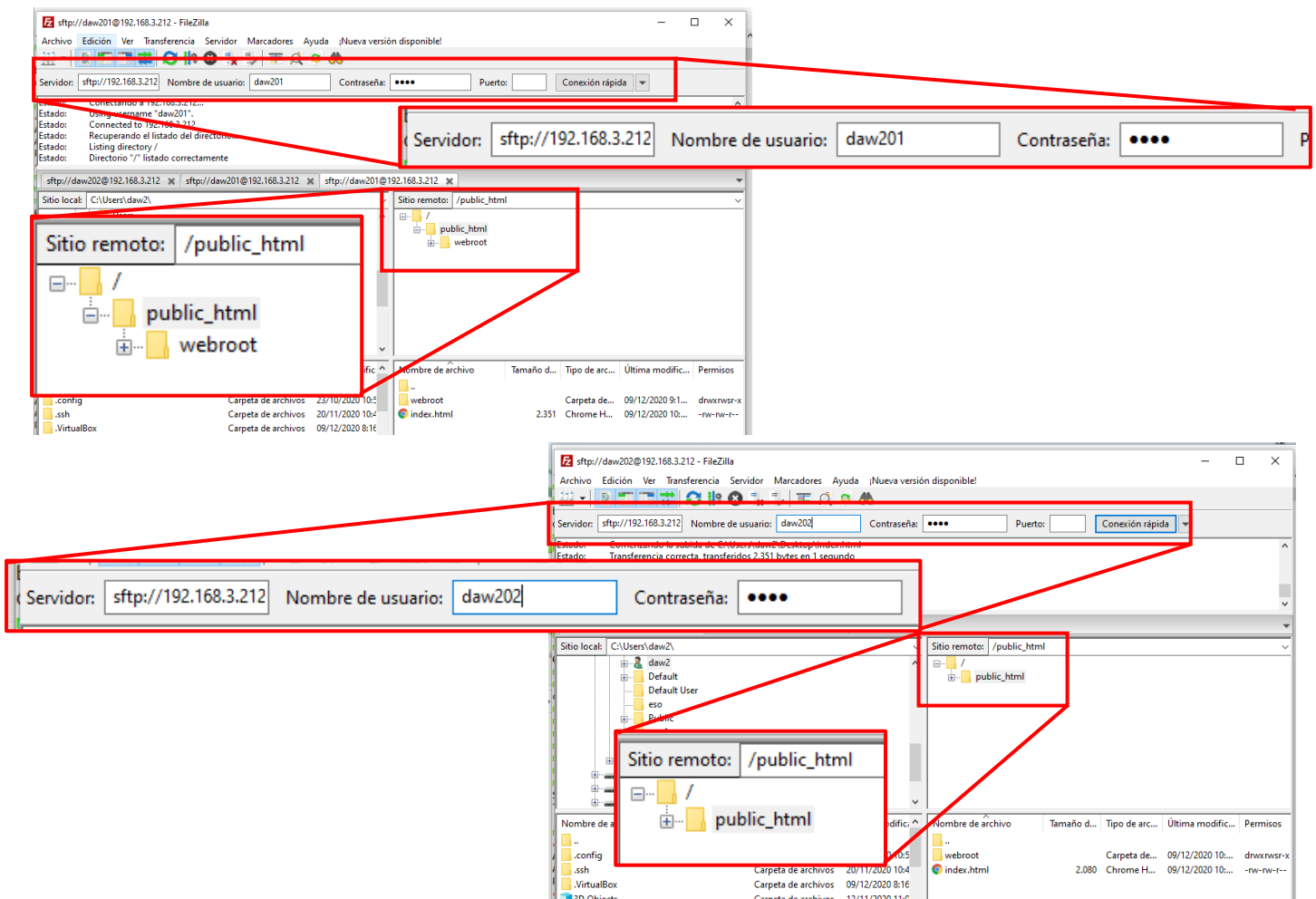
***sudo chmod 2775 -R /var/www/daw201/public\_html***

Propietario de public\_html

***sudo chown daw201:www-data -R /var/www/daw201/public\_html***

Con este mismo proceso crearemos daw202

Visualizaremos la carpeta public\_html y introduciremos un index para su visualización.



Todos los sitios virtuales tienen un archivo index.html con un mensaje de Bienvenida.

Este archivo lo crea y transfiere al servidor el usuario creado.

Los pasos que seguir para este procedimiento serán los siguientes:

1. Entraremos en el directorio `cd /etc/apache2/sites-available`
2. Realizaremos una copia del sitio `000-default.conf`, el nombre del nuevo sitio será significativo

**`sudo cp 000-default.conf daw201.conf`**

**`sudo cp 000-default.conf daw202.conf`**

```
miadmin@NAJUSED:/etc/apache2/sites-available$ sudo cp 000-default.conf daw201.conf
miadmin@NAJUSED:/etc/apache2/sites-available$ sudo nano daw201.conf
```

Editaremos el fichero que acabamos de crear, `sudo nano -----.conf`

Una vez dentro añadiremos 4 líneas:

```
ServerAdmin daw201@nereaa.local
ServerName daw201.nereaa.local
ServerAlias www.daw201.nereaa.local
DocumentRoot /var/www/daw201/public_html
```

En este enlace se puede ver los archivos con sus líneas.

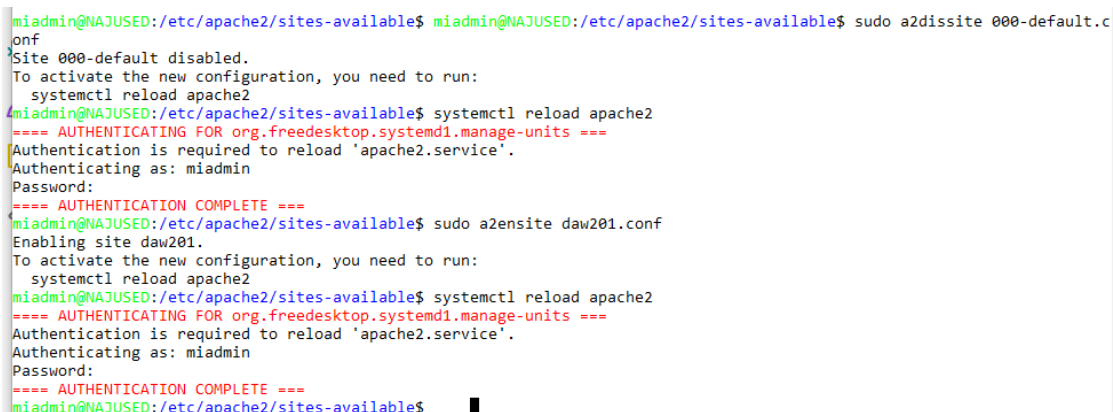
<https://github.com/IESSAUCES/SERVERDAW/tree/ramaNereaA/apache2/sites-available>

4. Se desactiva el sitio 000-default.conf

```
sudo a2dissite 000-default.conf
systemctl reload apache2
```

5. Posteriormente se activa el sitio nuevo que corresponda cada vez

```
sudo a2ensite daw201.conf
systemctl reload apache2 .....
```



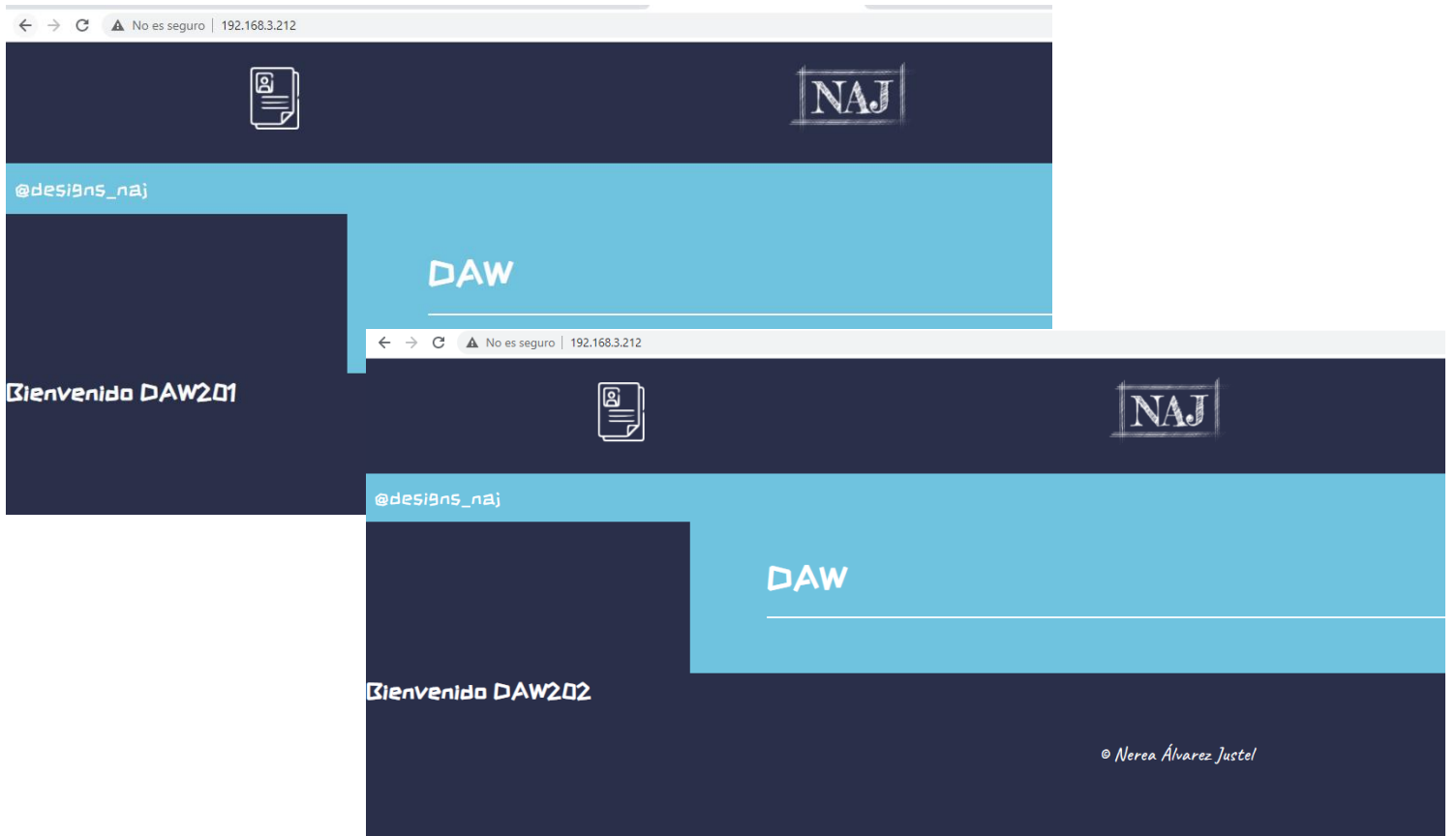
```
miadmin@NAJUSED:/etc/apache2/sites-available$ sudo a2dissite 000-default.conf
Site 000-default disabled.
To activate the new configuration, you need to run:
  systemctl reload apache2
miadmin@NAJUSED:/etc/apache2/sites-available$ systemctl reload apache2
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to reload 'apache2.service'.
Authenticating as: miadmin
Password:
==== AUTHENTICATION COMPLETE ====
miadmin@NAJUSED:/etc/apache2/sites-available$ sudo a2ensite daw201.conf
Enabling site daw201.
To activate the new configuration, you need to run:
  systemctl reload apache2
miadmin@NAJUSED:/etc/apache2/sites-available$ systemctl reload apache2
==== AUTHENTICATING FOR org.freedesktop.systemd1.manage-units ====
Authentication is required to reload 'apache2.service'.
Authenticating as: miadmin
Password:
==== AUTHENTICATION COMPLETE ====
miadmin@NAJUSED:/etc/apache2/sites-available$
```

6. La activación se verá en el directorio cd /etc/apache2/sites-enabled, saldrán de color azul

```
miadmin@NAJUSED:/etc/apache2/sites-enabled$ ls
daw201.conf
miadmin@NAJUSED:/etc/apache2/sites-enabled$
```

7. Reiniciamos el servidor

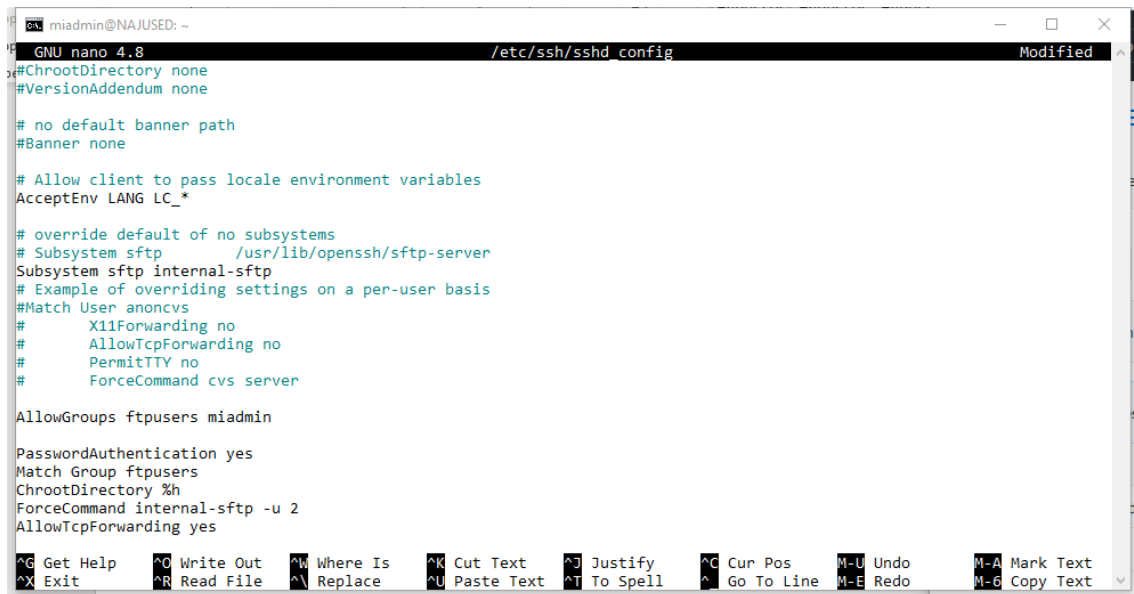
## 8. Comprobaremos





## 7. Control de acceso al servicio SSH, solo pueden acceder los usuarios que pertenecen al grupo ftpusers y el usuario miadmin.

Con el comando conseguiremos que solo sea accesible para miadmin y los usuarios enjaulados.



```
GNU nano 4.8 /etc/ssh/sshd_config Modified
#ChrootDirectory none
#VersionAddendum none

# no default banner path
#Banner none

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*

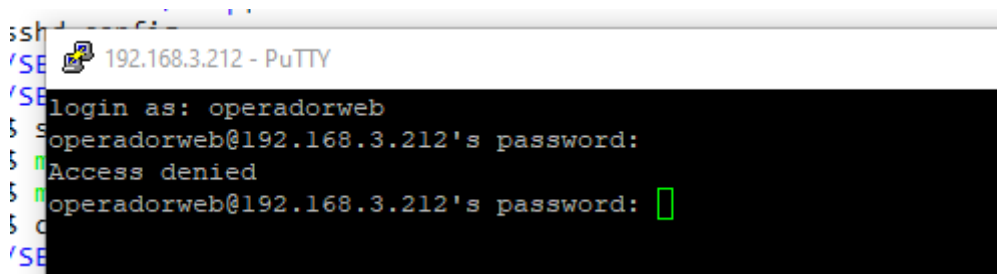
# override default of no subsystems
# Subsystem sftp /usr/lib/openssh/sftp-server
Subsystem sftp internal-sftp
# Example of overriding settings on a per-user basis
#Match User anoncvs
#       X11Forwarding no
#       AllowTcpForwarding no
#       PermitTTY no
#       ForceCommand cvs server

AllowGroups ftpusers miadmin

PasswordAuthentication yes
Match Group ftpusers
ChrootDirectory %h
ForceCommand internal-sftp -u 2
AllowTcpForwarding yes

^G Get Help  ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos    M-U Undo      M-A Mark Text
^X Exit      ^R Read File  ^N Replace    ^U Paste Text ^T To Spell   ^_ Go To Line  M-E Redo      M-G Copy Text
```

Comprobacion de la denegacion de acceso de operaci3nweb.



```
192.168.3.212 - PuTTY
login as: operatorweb
operatorweb@192.168.3.212's password:
Access denied
operatorweb@192.168.3.212's password: 
```

## 8. Investiga como controlar la cuota de disco asignada a cada usuario Linux.

El almacenamiento en disco se puede restringir mediante **la implementación de cuotas** de disco y de esta manera el **administrador** es **notificado** antes de que un usuario **consume mucho espacio en disco** o que una partición se llene.

Las cuotas se pueden configurar para **usuarios individuales** o para **grupos**. Este tipo de **flexibilidad** hace posible darle a cada usuario una **pequeña porción del disco** para que maneje sus **archivos personales**, mientras que se le permite tener **más espacio** para manejar los **proyectos** en los que estén trabajando o cuotas **más grandes**, asumiendo que a los **proyectos** se les de sus propios **grupos**.

Además, se puede configurar las cuotas no sólo para que controlen el número de bloques de disco, pero también el número **de inodes**. Debido a que los inodes son usados para contener **información relacionada a los archivos**, esto permite controlar el **número de archivos** que pueden ser **creados**.

El **RPM quota** debe estar instalado para implementar las cuotas de disco.

## CONCLUSIÓN

He realizado un estudio de los protocolos de transferencia de archivos y sus distintos tipos o variantes, estos tipos contienen diferentes características o son evoluciones con mejoras y mas compatibles con las necesidades actuales.

Aprendizaje de enjaulado de usuarios en “masa”, con estos scripts crearemos un gran número de usuarios para un sitio en concreto. Y al igual con los scripts de borrado, con ellos podremos borrar sitio y todos los usuarios creados.

## BIBLIOGRAFÍA

### Ejercicio 1

<https://www.blai.blog/2019/01/comparando-ftp-https-ftpes-sftp-fxp-tftp.html>

<https://blog.neothek.com/cuales-son-las-diferencias-entre-ftp-sftp-y-https/>

<https://www.smartfile.com/blog/comparison-of-ftp-https-sftp/>

### Ejercicio 2

<https://www.ionos.es/ayuda/hosting/configurar-y-gestionar-accesos-ftp/por-que-deberia-usar-sftp-y-https-en-lugar-de-ftp/>

<https://www.goanywhere.com/es/blog/sftp-vs-https-cuales-son-las-principales-diferencias>

<https://blog.ahierro.es/ftp-https-y-sftp-diferencias-ventajas-inconvenientes/>

<https://kinsta.com/es/base-de-conocimiento/ftp-vs-sftp/>

### Ejercicio 3

<https://www.nettix.com.pe/hosting/diferencia-entre-ftp-https-y-sftp>

### Repositorio

<https://github.com/IESSAUCES/SERVERDAW/tree/ramaNereaA>

### Ejercicio 7

<https://ostechnix.com/allow-deny-ssh-access-particular-user-group-linux/>

### Ejercicio 8

<http://web.mit.edu/rhel-doc/3/rhel-sag-es-3/ch-disk-quotas.html>

[https://www.linuxtotal.com.mx/index.php?cont=info\\_admon\\_018](https://www.linuxtotal.com.mx/index.php?cont=info_admon_018)

<https://www.ochobitshacenunbyte.com/2016/09/06/cuotas-de-disco-en-linux/>

