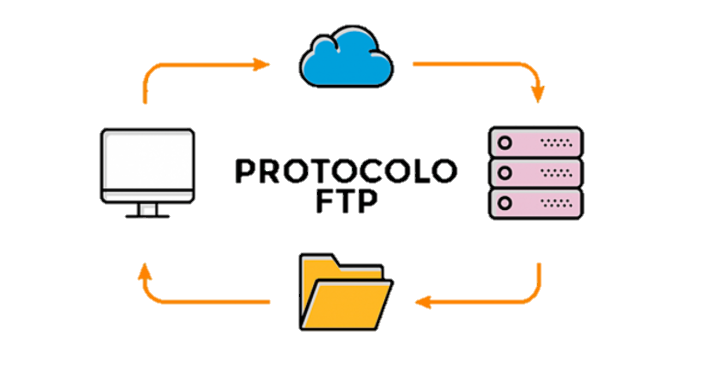
Manual LAMp y ftp

Sara del Pino Cabrera Sánchez 2º DAW





índice

[Cambiar nombre de Usuario 3](#_Toc118929521)

[Instalación Servidor LAMP 3](#_Toc118929522)

Apache……………………………………………………………………………………………………………………………….3

MySQL………………………………………………………………………………………………………………………………..5

PHP……………………………………..……………………………………………………………………………………………..5

Web por defecto…………………………………………………………………………………………………………………6

Configuración MySQL………………………………………………………………………………………………………….7

Instalación VPS……………………………………………………………………………………………………………………9

[Instalación PHPMyadmin 13](#_Toc118929523)

[Instalación HTTPS 14](#_Toc118929524)

Certificado SSL………………………………………………………………………………………………………………….14

Configuración……………………………………………………………………………………………………………………15

Instalación FTP 17

Logs………………………………………………………………………………………………………………………………….18

Modificar mensaje……………………………………………………………………………………………………………19

Carpetas por defecto………………………………………………………………………………………………………..19

Creación de grupos y usuarios………………………………………………………………………………………….20

Restringir usuarios……………………………………………………………………………………………………………22

FTP privado y anónimo……………………………………………………………………………………………………..23

Manual

# Cambiar nombre de Usuario

Para cambiar el nombre de usuario que se ve desde nuestra terminal, tendremos que cambiar los ficheros *hosts* y *hostname* que se encuentran en el directorio /etc. En ellos sobrescribiremos el antiguo nombre para poner el nuevo:

# Fichero hostname

sara\_cabrera

# Fichero hosts

127.0.1.1 sara\_cabrera

1

2

3

4

5



Debemos tener en cuenta reiniciar nuestra máquina tras aplicar estos cambios.

# Instalación Servidor LAMP

#### Apache

Para este tutorial es importante saber que haremos uso de permisos como administrador, así que es interesante hacer primero un:

1

$ sudo su

Primeramente, abriremos nuestra terminal para actualizar nuestro sistema y, tras eso, instalaremos Apache2.

Es importante saber que *update* actualiza el software dentro de la versión que tenemos definida, y *upgrade* actualiza el software a la versión más reciente.

1

2

3

4

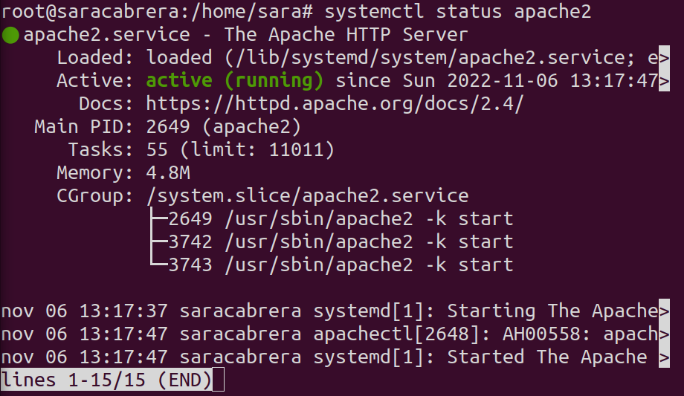
5

# Actualizamos el sistema

$ apt update

# Instalamos Apache

$ apt install apache2

Una vez instalado, podremos comprobar el estado de nuestro servidor Apache haciendo uso

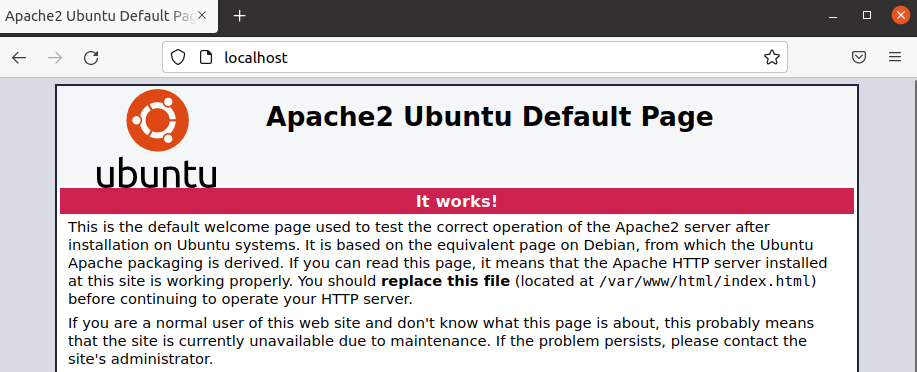
1

$ systemctl status apache2

de:

O también podemos acceder desde nuestro navegador

a localhost o 127.0.0.1



Ahora tendremos que activar el cortafuegos (firewall) de nuestra máquina:

1

2

3

4

5

6

7

8

# Activamos el FireWall

$ ufw enable

# Muestra aquellas Apps con el FireWall activado

$ ufw app list

# Evitamos que nos proteja de las peticiones al servidor apache

$ ufw allow in “Apache”



Otros comandos sobre FireWall que pueden sernos útiles:

# Podemos desactivar el FireWall con:

$ ufw disable

# Para saber el estado de nuestro FireWall podemos usar:

$ ufw status

1

2

3

4

5

En caso de que queramos acceder a nuestra máquina desde fuera necesitaremos nuestra ip pública. Esta podemos conseguirla de la siguiente manera:

1

2

3

4

5

6

# Vemos nuestra configuración ip

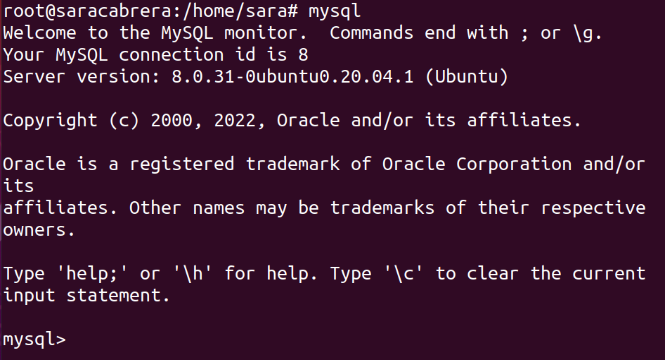
$ ip add

# Para mostrar nuestra ip pública:

$ apt install curl

$ curl http://icanhazip.com

#### MySQL

A continuación, pasaremos a instalar mysql-server (sin seguridad):

# Instalamos Mysql

$ apt install mysql-server

# Comprobamos que funcione correctamente

$ mysql

1

2

3

4

5

#### PHP

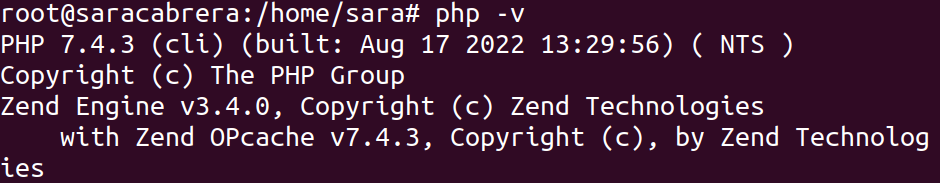
Ahora toca hacer la instalación de PHP en nuestro servidor, para ello instalaremos los módulos correspondientes:

1

2

# Instalamos los módulos necesarios

$ apt install php libapache2-mod-php php-mysql

Para comprobar que se ha instalado correctamente podremos preguntar por la versión de php que tenemos en nuestro servidor:

# Comprobamos la versión de php

$ php -v

1

2

#### Web por defecto

Para poder cambiar la página web que por defecto nos abre nuestro servidor tendremos que:

# Ir al directorio /var/www/html

$ cd /var/www/html

# Hacer una copia de seguridad de index.html

$ cp index.html index\_copia.html

# Modificamos el index.html a nuestro gusto.

1

2

3

4

5

6

7

También podemos crear un index.php y que éste se abra por defecto en nuestro servidor, para ello tendremos que cambiar la configuración de Apache de la siguiente forma:

Es interesante conocer las diferentes formas de abrir un fichero en Linux. Tenemos la opción de *nano*, esta nos abrirá el fichero desde la terminal. Por otro lado, tenemos la opción de *gedit*, esta abrirá nuestro fichero en un archivo txt aparte, siendo mucho más cómodo de modificar y leer.

# Nos movemos al directorio /etc/apache2/mods-enable

$ cd /etc/apache2/mods-enable

# Editamos el fichero dir.conf

$ gedit dir.conf

# Añadimos el index.php como primera opción

# Reseteamos el Servicio de Apache

$ systemctl reload apache2

1

2

3

4

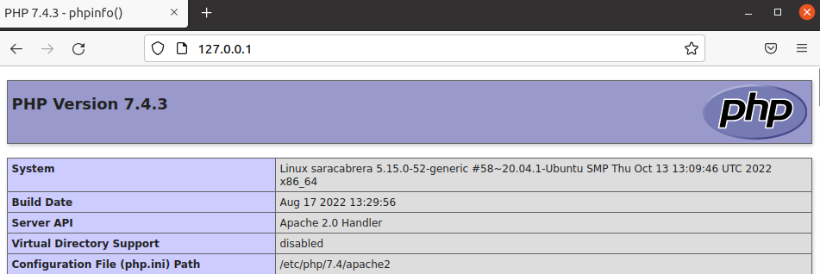
5

6

7

8

9

A continuación, añadiremos contenido de prueba al index.php y podremos visualizarlo en la web de nuestro servidor.

1

<?php phpinfo(); ?>

#### Configuración MySQL

Para configurar MySQL seguiremos los siguientes pasos (dentro de mysql):

CREATE DATABASE dbDPL;

# Creamos el usuario y le introducimos la contraseña

CREATE USER '*SaraC*'@'%' IDENTIFIED WITH MYSQL\_NATIVE\_PASSWORD BY '*1234*';

# Damos permisos al usuario creado sobre la base de datos

GRANT ALL ON dbDPL.\* TO '*SaraC*'@'%';

1

2

3

4

5

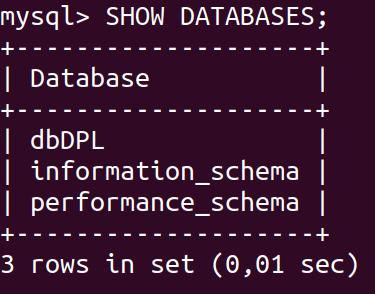
6

7

Una vez que hemos creado a nuestro usuario y le hemos dado los permisos correspondientes, iniciaremos mysql como dicho usuario:

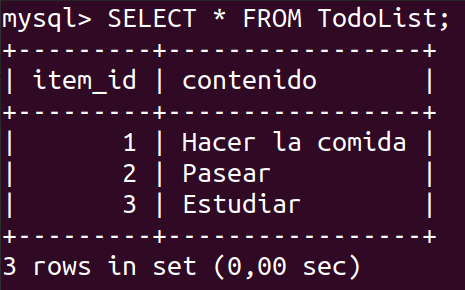
$ mysql -u SaraC -p

1

Una vez insertemos la contraseña correctamente nos dejará iniciar mysql. En caso de que queramos comprobar que efectivamente hemos creado bien la base de datos podemos mostrarla de la siguiente forma:

SHOW DATABASES;

1

A continuación, crearemos una tabla y le insertaremos datos:

USE dbDPL;

CREATE TABLE TodoList(

item\_id INT AUTO\_INCREMENT,

contenido VARCHAR(255),

PRIMARY KEY (item\_id)

);

INSERT INTO TodoList (contenido) VALUES

(“Hacer la comida”),

(“Pasear”),

(“Estudiar”);

SELECT \* FROM TodoList;

1

2

3

4

5

6

7

8

9

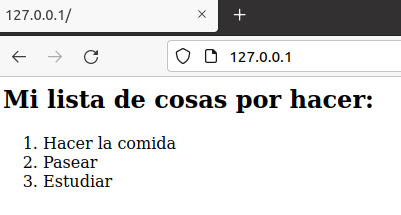
10

11

12

13

14

Y cambiaremos el index.php a lo siguiente:

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

<?php

$user="SaraC";

$password="1234";

$database="dbDPL";

$table="TodoList";

try {

$db = new PDO("mysql:host = localhost; dbname=$database", $user, $password);

echo "<h2> Mi lista de cosas por hacer: </h2><ol>";

foreach ($db -> query("SELECT contenido FROM $table") as $row) {

echo "<li>" . $row["contenido"] . "</li>";

} echo "</ol>";

} catch (PDOException $e) {

print "Error!" . $e -> getMessage() . "<br/>";

die();

}

?>

Podemos ver los permisos de un usuario en concreto haciendo uso de:

1

2

# Para ver los permisos de un usuario.

SHOW GRANTS FOR “SaraC”@”%”;

#### Instalación VPS

Debemos dirigirnos al directorio /var/www y crear una carpeta para nuestro proyecto:

1

2

3

4

5

6

# Creamos la carpeta y entramos en ella

$ mkdir proyectoDPL.son

$ cd proyectoDPL.son

# Creamos un fichero html

$ gedit index.html

La estructura de nuestro html quedará de la siguiente forma:

1

2

3

4

5

6

7

8

9

10

<!DOCTYPE html>

<html lang="es">

<head>

<meta charset="UTF-8">

<title>Proyecto DPL</title>

</head>

<body>

<h1>Proyecto DPL</h1>

</body>

</html>

Ahora iremos al directorio /etc/apache2/sites-available y haremos una copia del fichero 000-default.conf cambiándole el nombre:

$ cd /etc/apache2/sites-available

$ cp 000-default.conf proyectoDPL.son.conf

1

2

En el nuevo fichero tendremos que añadir lo siguiente:

1

2

3

4

5

6

7

8

9

10

11

12

13

14

<VirtualHost \*:*80*>

ServerName *proyectoDPL.son*

ServerAlias *www.proyectoDPL.son*

ServerAdmin *webmaster@dominioDPL.com*

DocumentRoot */var/www/proyectoDPL.son*

<Directory */var/www/proyectoDPL.son*>

*Options -Indexes +FollowSymLinks*

*AllowOverride All*

</Directory>

ErrorLog ${APACHE\_LOG\_DIR}/*proyectoDPL.son-error.log*

CustomLog ${APACHE\_LOG\_DIR}/*proyectoDPL-access.log combined*

</VirtualHost>



Es interesante saber como ver los puertos abiertos en nuestro servidor:

1

2

3

4

5

6

7

8

9

# Opción 1

$ netstat -tplugn

# En caso de querer ver el estado de un puerto concreto

$ netstat -tplugn | grep:*puerto*

# Opción 2

$ lsoft -i

# En caso de querer ver el estado de un puerto concreto

$ lsoft -i | grep:*puerto*

Para indicarle a Apache cual es el puerto por el que debe escuchar, iremos a /etc/apache2/ports.conf y añadiremos la línea Listen :*puerto.*

1

2

3

4

5

6

7

8

# Iremos al siguiente directorio

$ /etc/apache2

# Abriremos el fichero

$ gedit ports.conf

# Añadiremos la siguiente línea

Listen :*puerto*

Una vez tengamos todo esto, debemos habilitar nuestro sitio web y recargar apache para que se apliquen los cambios:

# Habilitamos el sitio

$ a2ensite proyectoDPL.son.conf

# Recargamos Apache

$ systemctl reload apache2

1

2

3

4

5



Es bueno conocer también la directiva para deshabilitar un sitio web:

1

2

# Deshabilitamos el sitio

$ a2dissite proyectoDPL.son.conf

Ahora iremos al directorio /etc/apache2 y abrimos el fichero de apache2.conf para añadir el ServerName de nuestro proyecto.

1

2

# Al final del fichero

ServerName 127.0.0.1

Reseteamos el servidor y comprobamos que todo esté correctamente:

1

2

3

4

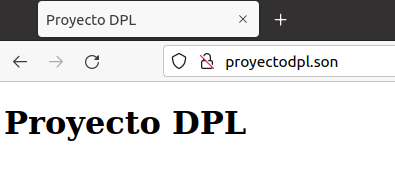
5

# Reseteamos el servidor

$ systemctl restart apache2

# Comprobamos que la configuración esté bien

$ apachectl configtest

Por último, debemos direccionar nuestra ip local al nuevo dominio que hemos creado, para ello modificamos el fichero hosts dentro del directorio /etc:

1

2

# Al principio del fichero

127.0.0.1 proyectoDPL.son

Ahora ya podremos acceder a *http://proyectoDPL.son* y comprobar que funciona correctamente.

En caso de tener algún fallo, podremos verlos desde el fichero proyectoDPL.son-error.log dentro del directorio /var/log/apache2:

1

2

3

4

5

# Para ver las últimas 10 líneas del fichero

$ tail -10 proyectoDPL.son-error.log

# Para ver las primeras 10 líneas del fichero

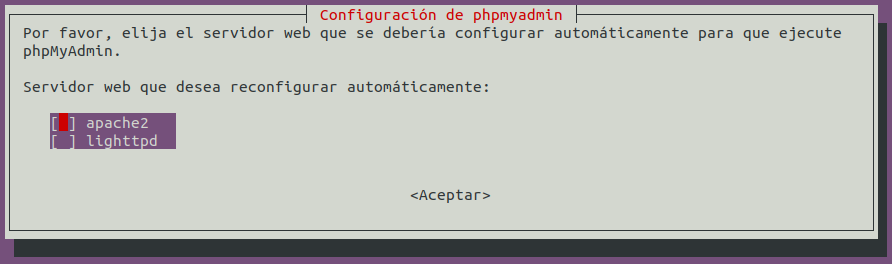
$ head -10 proyectoDPL.son-error.log

# Instalación PHPMyadmin

Para la instalación de PHPMyadmin tendremos que seguir los siguientes pasos:

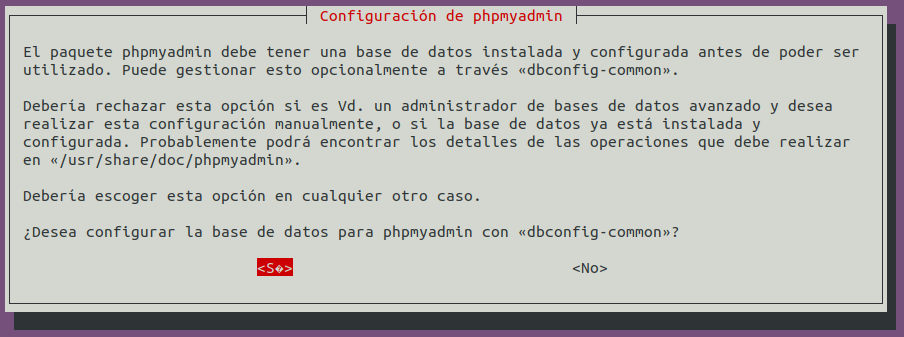
1

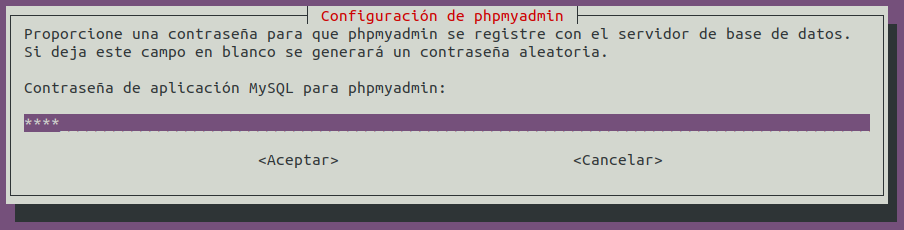
$ apt install phpmyadmin php-mbstring php-zip php-gd php-json php-curl

Aquí se nos abrirán diferentes desplegables:

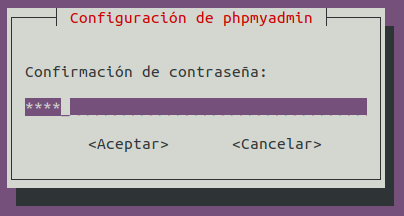
**2**

**1**

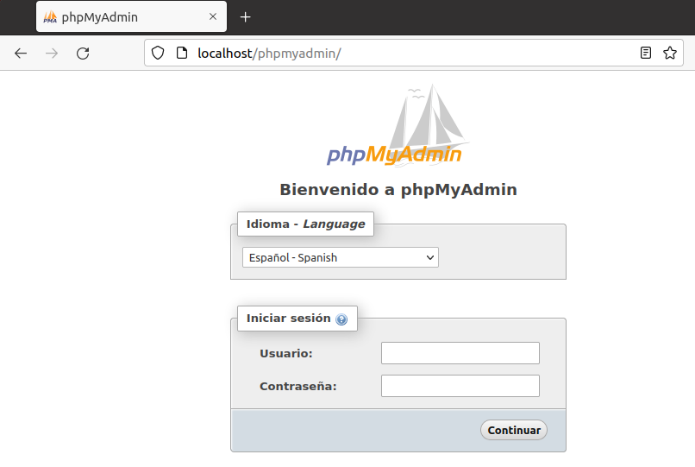




**3**



**4**

Y tras rellenarlo continuamos:

1

2

3

4

5

6

7

8

9

10

11

12

# Vamos a la carpeta de configuración del proyecto y abrimos apache2.conf

$ cd /etc/apache2

$ gedit apache2.conf

# Añadimos el directorio de phpmyadmin al final del fichero apache2.conf

Include /etc/phpmyadmin/apache.conf

# Habilitamos PHP y PHPMyAdmin

$ phpenmod mbstring

# Reseteamos apache

$ systemctl restart apache2

$ apt install phpmyadmin php-mbstring php-zip php-gd php-json php-curl

# apache si, contraseña que queramos

# Nos movemos a la carpeta de Configuración del Proyecto

$ cd /etc/apache2

# Añadimos el directorio de phpmyadmin al final del fichero

$ gedit apache2.conf

# Escribimos:

# Include /etc/phpmyadmin/apache.conf

# Habilitamos PHP y PHPMyAdmin

$ phpenmod mbstring

# Reseteamos el Servidor Apache

$ systemctl restart apache2

1

2

3

4

5

6

# Instalación HTTPS

Para instalar HTTPS en nuestras páginas primero tendremos que desactivar el cortafuegos:

1

$ ufw disable

A continuación, añadiremos el módulo SSL:

1

2

3

4

5

# Habilitamos el módulo ssl

$ a2enmod ssl

# Reseteamos apache

$ systemctl restart apache2

#### Certificado SSL

Ahora generaremos el certificado SSL firmado por nosotros:

$ openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt

# Rellenamos los campos que nos saldrán a continuación:

Country Name (2 letter code) [XX]: *ES*

State or Province Name (full name) []: *LPA*

Locality Name (eg, city) [Default City]: *LPA*

Organization Name (eg, company) [Default Company Ltd*]: sarcabsan.sl*

Organizational Unit Name (eg, section) []: *section*

Common Name (eg, your name or your server's hostname) []: *www.proyecto.son*

Email Address []: *correo*

1

2

3

4

5

6

7

8

9

10

#### Configuración

Cambiamos la configuración en el archivo proyectoDPL.son.conf dentro del directorio /etc/apache2/sites-available:

1

2

3

4

5

6

7

8

9

10

11

# Abrimos el fichero

$ gedit proyectoDPL.son.conf

# Cambiamos el puerto por el 443 (Puerto por defecto de HTTPS)

<VirtualHost \*:*443*>

...

# En el fichero escribimos esto al final del todo

SSLEngine *on*

SSLCertificateFile */etc/ssl/certs/apache-selfsigned.crt*

SSLCertificateKeyFile */etc/ssl/private/apache-selfsigned.key*

</VirtualHost>

Deshabilitamos el proyecto y reseteamos apache:

# Deshabilitamos el proyecto

$ a2dissite proyectoDPL.son.conf

# Reseteamos apache

$ systemctl restart apache2

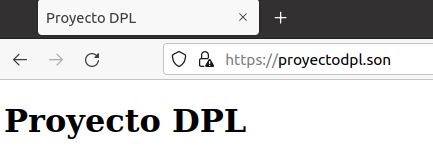
1

2

3

4

5

Volvemos a habilitar el proyecto, comprobamos que la configuración esté correcta y recargamos apache:

1

2

3

4

5

6

7

8

9

# Habilitamos el proyecto

$ a2ensite proyectoDPL.son.conf

# Comprobamos que la configuración esté bien hecha

$ apache2ctl configtest

# En caso de que todo esté correctamente nos saldrá un mensaje de “Syntax OK”

# Recargamos apache

$ systemctl reload apache2

En caso de que esto no nos funcione podemos probar a abrir los puertos 443 y 80:

1

2

# Abrimos los puertos 443 y 80

$ ufw allow “Apache Full”

Por último, es interesante redireccionar el http al https, para ello iremos al fichero de configuración proyectoDPL.son.conf en el directorio /etc/apache2/sites-available:

1

2

3

4

5

6

7

8

# Añadimos un virtualhost para el puerto 80 fuera del que ya tenemos

<VirtualHost \*:*80*>

ServerName *proyectoDPL.son*

Redirect */ https://proyectoDPL.son*

</VirtualHost>

# Recargamos apache

$ systemctl reload apache2

# Instalación FTP



Primeramente, es interesante conocer los principales comandos de FTP:

1

2

3

4

5

6

7

8

9

10

11

12

help o ? -> enumera todos los comandos de FTP disponibles.

cd -> cambia el directorio en la máquina remota.

lcd -> cambiar el directorio en la máquina local.

ls -> muestra los nombres de los archivos y directorios en el directorio remoto actual.

mkdir -> crea un nuevo directorio dentro del directorio remoto.

pwd -> imprime el directorio de trabajo actual en la máquina remota.

delete -> elimina un archivo en el directorio remoto actual.

rmdir -> elimina un directorio en el directorio remoto actual.

get -> copia un archivo del servidor remoto a la máquina local.

mget -> permite copiar múltiples archivos del servidor remoto a la máquina local.

put -> copia un archivo de la máquina local a la máquina remota.

mput -> copia un archivo de la máquina remota a la máquina local.

Para instalar el FTP haremos uso de las siguientes directivas:

1

2

3

4

5

# Actualizamos el software del equipo

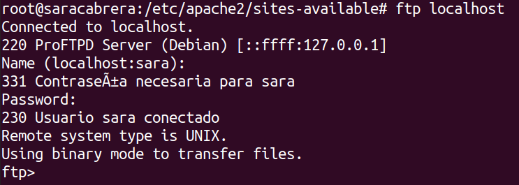
$ apt update

# Instalamos el módulo

$ apt install proftpd

Tras la instalación se nos habrán generado una serie de ficheros de configuración, los más importantes son:

* proftpd.conf: archivo de configuración de proftpd, se encuentra en el directorio /etc/proftpd/
* ftpusers: fichero 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. Se encuentra en el directorio /etc

Para conectarnos al FTP lo haremos de la siguiente forma:

# ftp “dirección\_ip”

$ ftp localhost

1

2

Al conectarnos nos pedirá un usuario y una contraseña que, por defecto, serán las de nuestro equipo.



Algunas de las directivas más básicas a la hora de configurar nuestro proftpd.conf son:

* ServerName: nombre de nuestro servidor.
* DeferWelcome: con esto activado, no mostraremos información sobre nuestro equipo hasta que el usuario haga login.
* ShowSymlinks: habilita/deshabilita los enlaces simbólicos.
* TimeoutIdle: configura la cantidad máxima de segundos que proftpd permitirá a los clientes permanecer conectados sin recibir ningún dato.
* Port: puerto de nuestro servidor (21 por defecto).
* TransferLog: especifica la ruta al registro de transferencia.
* SystemLog: ruta del fichero en donde guardamos los logs.

#### Logs

Para mostrar el contenido de los logs de proftpd iremos a la ruta /var/log/proftpd y ejecutaremos el fichero proftpf.log:

1

2

3

4

5

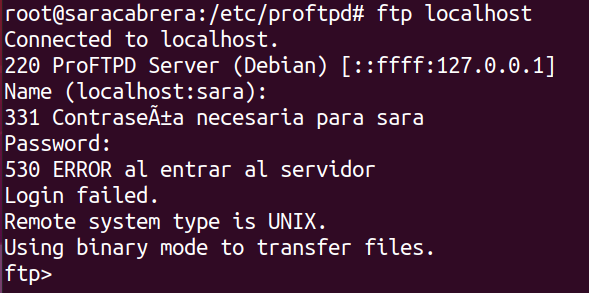
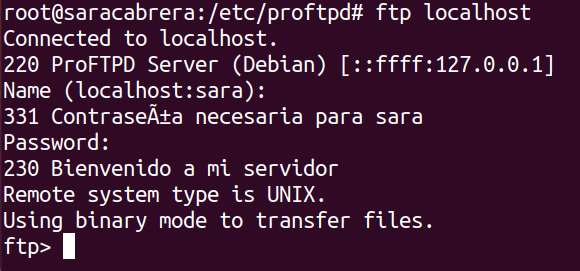
# Vamos al siguiente directorio:

$ cd /var/log/proftpd

# Muestra los logs

$ cat proftpd.log

#### Modificar mensaje

Para modificar el mensaje de bienvenida y de error iremos a la ruta /etc/proftpd y editaremos el fichero proftpd.conf:

1

2

3

4

5

6

7

8

9

10

11

12

# Vamos al siguiente directorio:

$ cd /etc/proftpd

# Abrimos el fichero proftpd

$ gedit proftpd.conf

# Insertamos las siguientes líneas al final del fichero:

AccessGrantMsg *"Bienvenido a mi servidor"*

AccessDenyMsg *"ERROR al entrar al servidor"*

# Reseteamos el FTP

$ service proftpd restart

#### Carpetas por defecto

Existe una directiva llamada DefaultRoot con la que podemos definir a que carpeta entrarán, por defecto, nuestros usuarios al FTP (va colocada al final del fichero proftpd.conf del directorio /etc/proftpd):

# Sintaxis de la directiva:

DefaultRoot [directorio] [grupo de usuarios que tendrán este directorio por defecto] [grupo de usuarios a los que no se les aplicará el directorio]

# Por defecto definimos home

DefaultRoot ~

# Por defecto definimos /var/ftpXX

DefaultRoot /var/ftpXX

1

2

3

4

5

6

7

8

#### Creación de grupos y usuarios

Para la creación de usuarios usaremos:

Podemos eliminar un usuario con *userdel* seguido del nombre del usuario a eliminar.

1

2

# Creamos al usuario user1dpl

$ adduser user1dpl



Debemos tener en cuenta las diferencias entre adduser y useradd:

* **useradd**: crea un nuevo usuario, pero no le crea su respectiva carpeta en /home. En caso de que queramos crearla tendremos que indicarlo con:

1

$ useradd -m usernuevo

* **adduser**: crea un nuevo usuario y su respectiva carpeta en /home. La desventaja es que este comando no es compatible con todas las distribuciones de Linux.

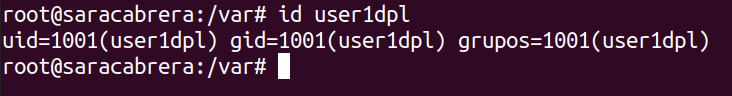
Podemos comprobar que un usuario ha sido creado correctamente haciendo uso de:

# Muestra todos los usuarios de nuestro equipo

$ cat /etc/passwd

1

2

También podemos conocer la configuración de un usuario en concreto con:

1

$ id user1dpl

Esto nos indicará, entre otras cosas, a que grupos pertenece el usuario.

Para crear un grupo y asignárselo a un usuario seguiremos los siguientes pasos:

Podemos eliminar un grupo con *delgroup* seguido del nombre del grupo a eliminar.

1

2

3

4

5

# Creamos el grupo

$ addgroup grupo1

# Añadimos a un usuario al grupo

$ usermod -a -G grupo1 user1dpl

También podemos cambiar los permisos y los grupos de diferentes carpetas, para ello podemos hacer uso de los siguientes comandos:

1

2

3

4

5

6

7

8

# Para ver los grupos y permisos sobre una serie de carpetas:

$ ls -l

# Para poner a un grupo como propietario de una carpeta:

$ chgrp grupo1 ftpXX

# Para darle todos los permisos al grupo sobre dicha carpeta:

$ chmod 775 ftpXX

Para la hora de dar permisos sobre diferentes ficheros es muy importante conocer cómo se establecen estos:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| r | w | x | Número | Permisos |
| 0 | 0 | 0 | 0 | --- |
| 0 | 0 | 1 | 1 | --x |
| 0 | 1 | 0 | 2 | -w- |
| 0 | 1 | 1 | 3 | -wx |
| 1 | 0 | 0 | 4 | r-- |
| 1 | 0 | 1 | 5 | r-x |
| 1 | 1 | 0 | 6 | rw- |
| 1 | 1 | 1 | 7 | rwx |

#### Restringir usuarios

Para poder restringir a ciertos usuarios a una única carpeta, tendremos que ir al directorio /etc/proftpd y modificar el fichero proftpd.conf añadiendo al final:

DefaultRoot */var/ftpXX grupo1*

# Reseteamos el FTP

$ service proftpd restart

1

2

3

4

Una vez hecho esto, podremos iniciar sesión con un usuario que pertenezca a dicho grupo y comprobaremos que se ha hecho todo correctamente.

Para permitir o denegar usuarios en nuestro FTP tendremos que modificar el fichero proftpd.conf de nuestro directorio /etc/proftpd:

# Añadimos al final de fichero:

<Limit LOGIN>

AllowUser *user1dpl*

DenyALL

</Limit>

1

2

3

4

5

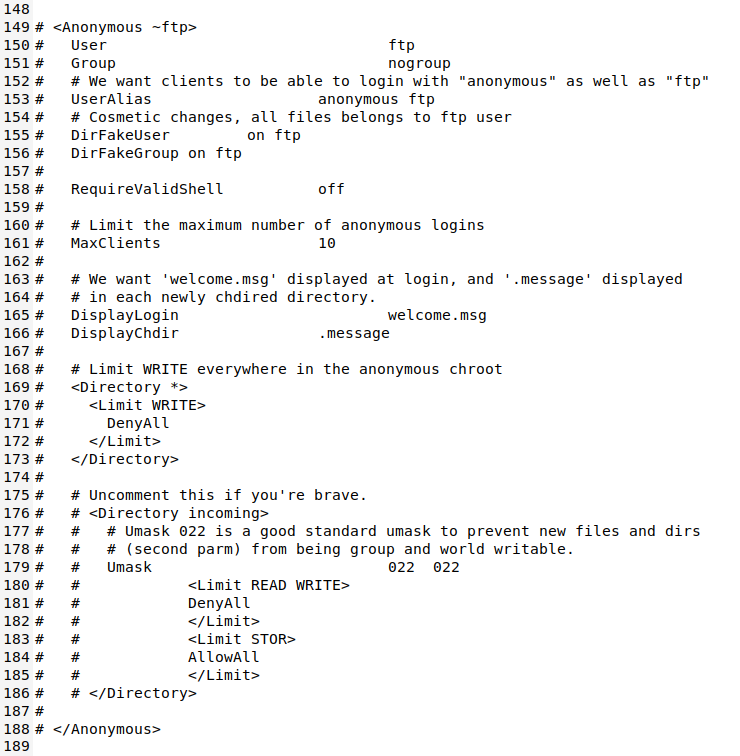


Estos son los diferentes parámetros que podremos usar:

* AllowUser [*nombre del usuario*]: permitimos el acceso a un usuario en específico.
* DenyUser [*nombre del usuario*]: bloqueamos el acceso a un usuario en específico.
* DenyAll: bloqueamos a todos los usuarios (salvo los que permitamos con AllowUser).
* AllowAll: permitimos que los usuarios como “anonymous” se conecten al FTP.

#### FTP privado y anónimo

Para configurar nuestro FTP como privado y anónimo tendremos que descomentar las siguientes líneas del fichero proftpd.conf de nuestro directorio /etc/proftpd:



Y con esto, si entramos como usuario anónimo veremos que no nos deja hacer nada.