



# Servicio de Red e Internet

Volodimir Yarmash Yarmash

# Índice

<b>Vamos a instalar un servidor web interno para un instituto. Se Pide:</b>	<b>2</b>
-Instalación del servidor web apache. Usaremos dos dominios mediante el archivo hosts: centro.intranet y departamentos.centro.intranet. El primero servirá el contenido mediante wordpress y el segundo una aplicación en python	2
Modificamos el fichero hosts	3
Creamos el archivo de configuración apache	4
Comprobamos	5
-Activar los módulos necesarios para ejecutar php y acceder a mysql	6
-Instala y configura wordpress	6
Comprobamos de la instalación	7
Creamos una base de datos con Mysql	7
Configuración Wordpress	8
-Activar el módulo "wsgi" para permitir la ejecución de aplicaciones Python	13
-Crea y despliega una pequeña aplicación python para comprobar que funciona correctamente	13
-Adicionalmente protegeremos el acceso a la aplicación python mediante autenticación	14
-Instala y configura awstat	15
-Instala un segundo servidor de tu elección (nginx, lighttpd) bajo el dominio "servidor2.centro.intranet". Debes configurarlo para que sirva en el puerto 8080 y haz los cambios necesarios para ejecutar php. Instala phpmyadmin	17

Vamos a instalar un servidor web interno para un instituto. Se Pide:

-Instalación del servidor web apache. Usaremos dos dominios mediante el archivo hosts: centro.intranet y departamentos.centro.intranet. El primero servirá el contenido mediante wordpress y el segundo una aplicación en python

Para instalar Apache2, necesitamos usar el comando `sudo apt update` para comprobar si hay alguna actualización importante.

Ahora introducimos `sudo apt install apache2` y esperamos a que se instale.

```
mario@mario-VirtualBox:~$ sudo apt update
[sudo] contraseña para mario:
Obj:1 http://es.archive.ubuntu.com/ubuntu jammy InRelease
Des:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu jammy-backports InRelease [99,8 kB]
Descargados 324 kB en 1s (262 kB/s)
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se pueden actualizar 28 paquetes. Ejecute «apt list --upgradable» para verlos.
mario@mario-VirtualBox:~$ sudo apt install apache2
```

Agregamos apache en el firewall para que no ocurran problemas.

```
user@user:~$ sudo ufw allow apache
[sudo] contraseña para user:
Reglas actualizadas
Reglas actualizadas (v6)
```

Comprobamos el status de Apache2 con el comando `sudo systemctl status apache2`

```

user@user:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: v
   Active: active (running) since Sun 2024-12-08 16:18:35 CET; 59min ago
     Docs: https://httpd.apache.org/docs/2.4/
  Process: 940 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUC>
 Main PID: 1004 (apache2)
    Tasks: 55 (limit: 3454)
   Memory: 7.7M (peak: 8.1M)
      CPU: 352ms
   CGroup: /system.slice/apache2.service
           └─1004 /usr/sbin/apache2 -k start
             └─1006 /usr/sbin/apache2 -k start
               └─1007 /usr/sbin/apache2 -k start

```

## Modificamos el fichero hosts

introducimos `sudo nano /etc/hosts` e introducimos los dominios de esta forma:

```

GNU nano 7.2 /etc/hosts *
127.0.0.1 localhost
127.0.1.1 user
127.0.0.1 centro.intranet
127.0.0.1 departamentos.centro.intranet

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

```

## Creamos el archivo de configuración apache

Introducimos el comando `cd /etc/apache2/sites-available` para posicionarnos en el directorio necesario.

Introducimos `sudo nano centro.intranet.conf`

Dentro tenemos que introducir esto, podemos sacarlo del archivo 000-default.conf y pegarlo.

```
user@user: /etc/apache2/sites-available
GNU nano 7.2 centro.intranet.conf *
<VirtualHost *:80>
  ServerName centro.intranet
  DocumentRoot /var/www/centro.intranet
  <Directory /var/www/centro.intranet>
    AllowOverride All
  </Directory>
</VirtualHost>
```

Ahora tenemos que crear el fichero para el otro dominio

Introducimos `sudo nano departamentos.centro.intranet.conf`

Introducimos este texto

```
GNU nano 7.2 departamentos.centro.intranet.conf *
<VirtualHost *:80>
  ServerName departamentos.centro.intranet
  DocumentRoot /var/www/departamentos.centro.intranet
  <Directory /var/www/departamentos.centro.intranet>
    AllowOverride All
  </Directory>
</VirtualHost>
```

Hacemos el ensite con el comando `sudo a2ensite (dominio)` y reiniciamos apache con el comando `sudo systemctl reload apache2`.

```
user@user:/etc/apache2/sites-available$ sudo a2ensite centro.intranet.conf
[sudo] contraseña para user:
Enabling site centro.intranet.
To activate the new configuration, you need to run:
  systemctl reload apache2
user@user:/etc/apache2/sites-available$ sudo a2ensite departamentos.centro.intranet.conf
Enabling site departamentos.centro.intranet.
To activate the new configuration, you need to run:
  systemctl reload apache2
user@user:/etc/apache2/sites-available$ sudo systemctl reload apache2
user@user:/etc/apache2/sites-available$
```

Creamos los directorios en la ruta `var/www`

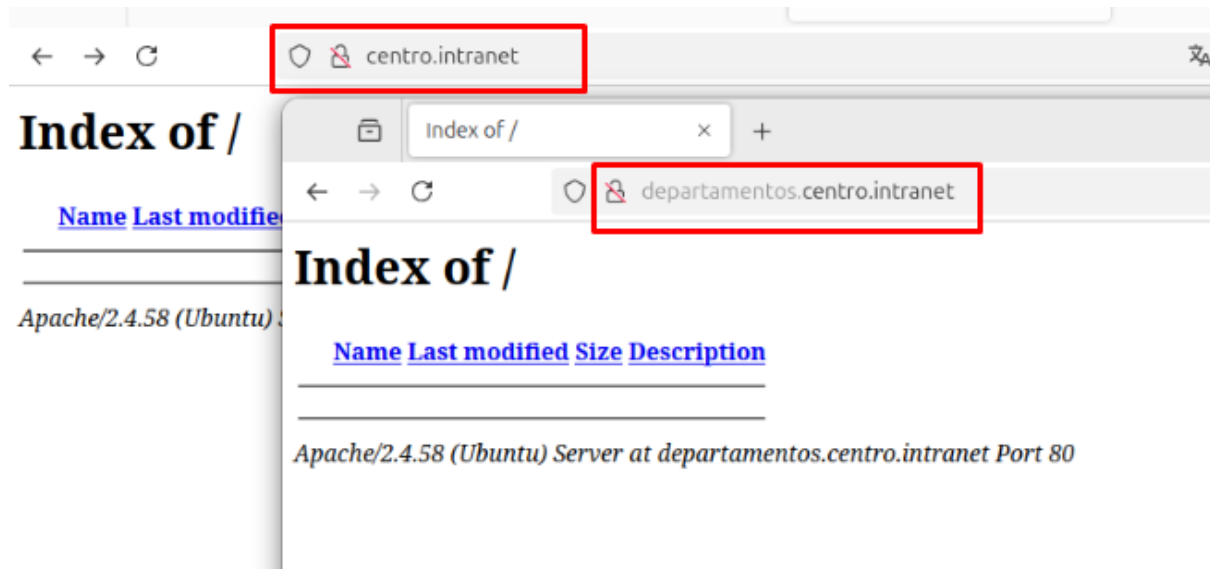
Accedemos al directorio usando el comando `cd /var/www`

Creamos el directorio con `sudo mkdir centro.intranet` y el segundo `sudo mkdir departamentos.centro.intranet`.

```
user@user:/etc/apache2/sites-available$ cd
user@user:~$ cd /var/www/
user@user:/var/www$ mkdir centro.intranet
mkdir: no se puede crear el directorio «centro.intranet»: Permiso denegado
user@user:/var/www$ sudo mkdir centro.intranet
user@user:/var/www$ sudo mkdir departamentos.centro.intranet
user@user:/var/www$
```

## Comprobamos

Para comprobar si hemos hecho bien la configuración, tenemos que meternos en la barra de búsqueda en cualquier buscador e introducir los dominios escribiendo por delante http://



## -Activar los módulos necesarios para ejecutar php y acceder a mysql

Entramos en el terminal e instalamos todo lo necesario con estos comandos

```
sudo apt install php
sudo apt install libapache2-mod-php
sudo apt install php-mysql
sudo apt install mysql-server
```

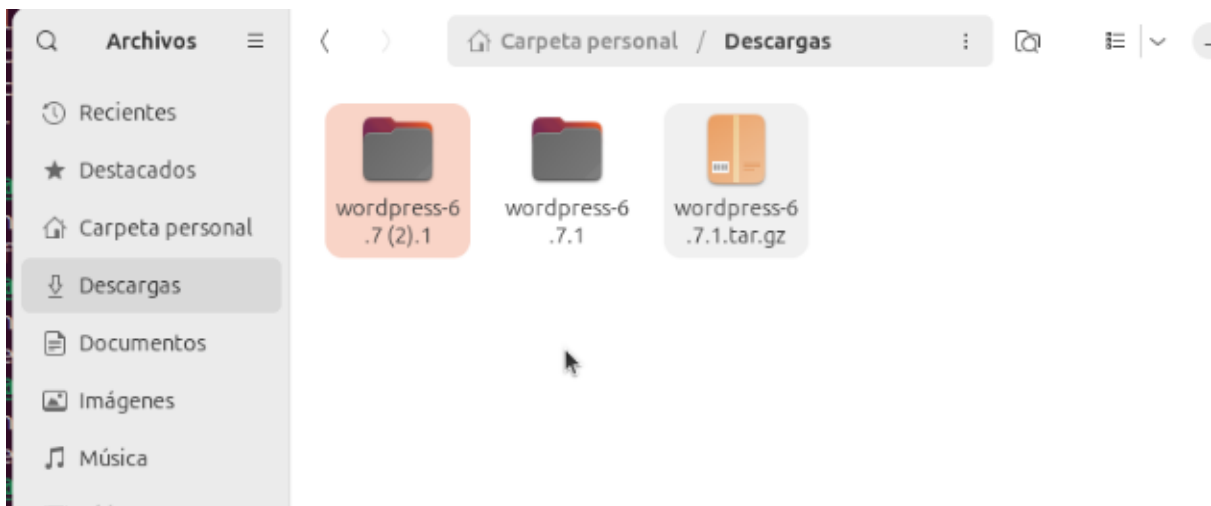
Y hacemos un restart con `sudo systemctl restart apache2`.

Con el comando `sudo ufw app list`, podemos ver si la instalación se ha realizado bien.

```
user@user:/var/www$ sudo ufw app list
Aplicaciones disponibles:
  Apache
  Apache Full
  Apache Secure
  CUPS
user@user:/var/www$
```

## -Instala y configura wordpress

Nos metemos en la página oficial [www.wordpress.com](http://www.wordpress.com) y descargamos wordpress. Descomprimos el archivo.



Nos situamos en el directorio de descargas, hacemos una copia a el directorio /var/www/html y añadimos todos los permisos con chmod.

```
user@user:~$ cd /home/user/
.cache/ Descargas/ Escritorio/ .local/ Plantillas/ snap/ Videos/
.config/ Documentos/ Imágenes/ Música/ Público/ .ssh/
user@user:~$ cd /home/user/
.cache/ Descargas/ Escritorio/ .local/ Plantillas/ snap/ Videos/
.config/ Documentos/ Imágenes/ Música/ Público/ .ssh/
user@user:~$ cd /home/user/Descargas/
user@user:~/Descargas$ cp wordpress-6.7
wordpress-6.7 (2).1/ wordpress-6.7.1/ wordpress-6.7.1.tar.gz
user@user:~/Descargas$ cp wordpress-6.7
wordpress-6.7 (2).1/ wordpress-6.7.1/ wordpress-6.7.1.tar.gz
user@user:~/Descargas$ cp wordpress-6.7 -R /var/www/html/
cp: no se puede efectuar 'stat' sobre 'wordpress-6.7': No existe el archivo o el directorio
user@user:~/Descargas$ cp wordpress-6.7.1/ -R /var/www/html/
cp: no se puede crear el directorio '/var/www/html/wordpress-6.7.1': Permiso denegado
user@user:~/Descargas$ sudo cp wordpress-6.7.1/ -R /var/www/html/
user@user:~/Descargas$ sudo chmod 777 -R /var/www/html/wordpress-6.7.1/
```

Comprobamos de la instalación.

Nos situamos en el buscador e introducimos localhost/wordpress-6.7.1



## Creamos una base de datos con Mysql

Introducimos el comando `sudo mysql -u root -p` y con sql creamos una base de datos usando el comando `create database wordpress;` salimos con `exit.`

```
user@user:~/Descargas$ sudo mysql -u root -p
[sudo] contraseña para user:
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.40-0ubuntu0.24.04.1 (Ubuntu)

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database wordpress;
Query OK, 1 row affected (0,88 sec)

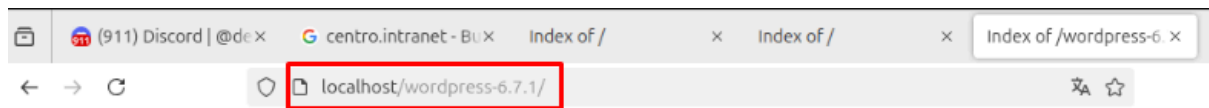
mysql> exit
Bye
user@user:~/Descargas$
```

## Configuración Wordpress

Entramos en <http://localhost/wordpress-6.7.1>

Pulsamos la carpeta de wordpress y después en el botón azul de la página que se sitúa abajo(Lets Go).

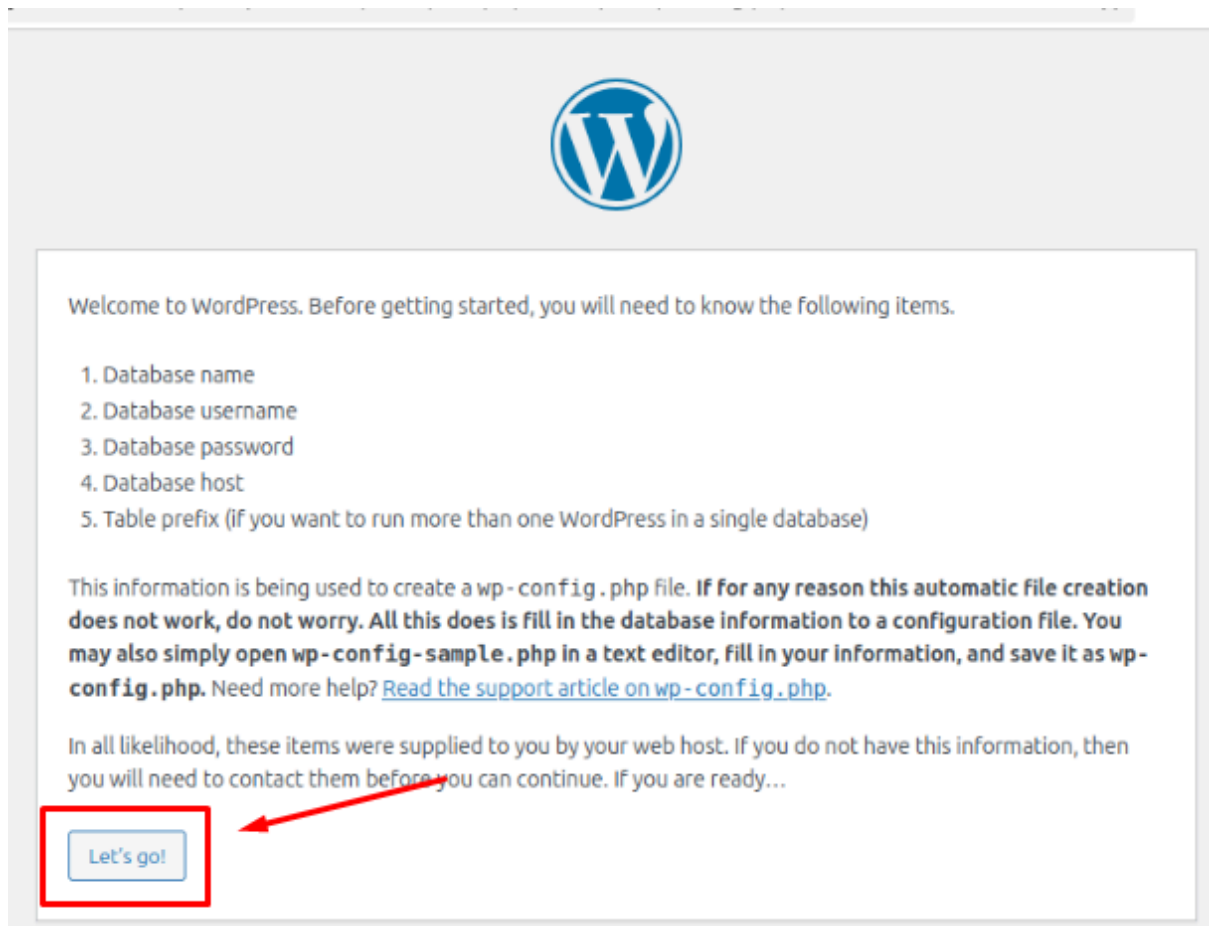




## Index of /wordpress-6.7.1

Name	Last modified	Size	Description
<a href="#">Parent Directory</a>	-	-	-
<a href="#">wordpress/</a>	2024-12-08 20:34	-	-

Apache/2.4.58 (Ubuntu) Server at localhost Port 80



Introducimos la información de autenticación, pulsamos submit.



Below you should enter your database connection details. If you are not sure about these, contact your host.

**Database Name**

wordpress

The name of the database you want to use with WordPress.

**Username**

user

Your database username.

**Password**

user

Your database password.

 Hide

**Database Host**

localhost

You should be able to get this info from your web host, if localhost does not work.

**Table Prefix**

wp\_

If you want to run multiple WordPress installations in a single database, change this.

Submit

Nos envía a la siguiente página.  
Rellenamos la información.

## Welcome

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

## Information needed

Please provide the following information. Do not worry, you can always change these settings later.

<b>Site Title</b>	<input type="text" value="centro"/>
<b>Username</b>	<input type="text" value="user"/> <small>Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.</small>
<b>Password</b>	<input type="password" value="user"/> <a href="#">Hide</a> <div>Very weak</div> <p><b>Important:</b> You will need this password to log in. Please store it in a secure location.</p>
<b>Confirm Password</b>	<input checked="" type="checkbox"/> Confirm use of weak password
<b>Your Email</b>	<input type="text" value="vyaryar541@g.educaand.es"/> <small>Double-check your email address before continuing.</small>
<b>Search engine</b>	<input type="checkbox"/> Discourage search engines from indexing this site

Pues listo, ya tenemos wordpress.



## Success!

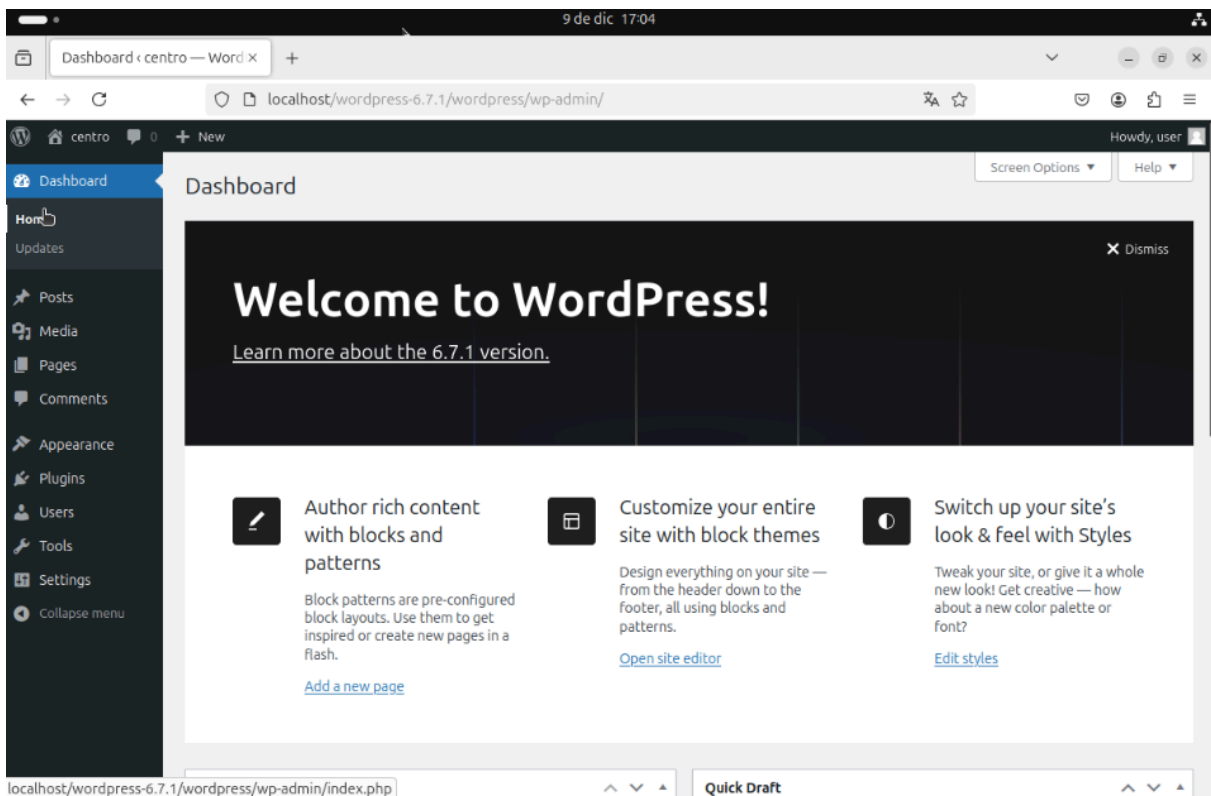
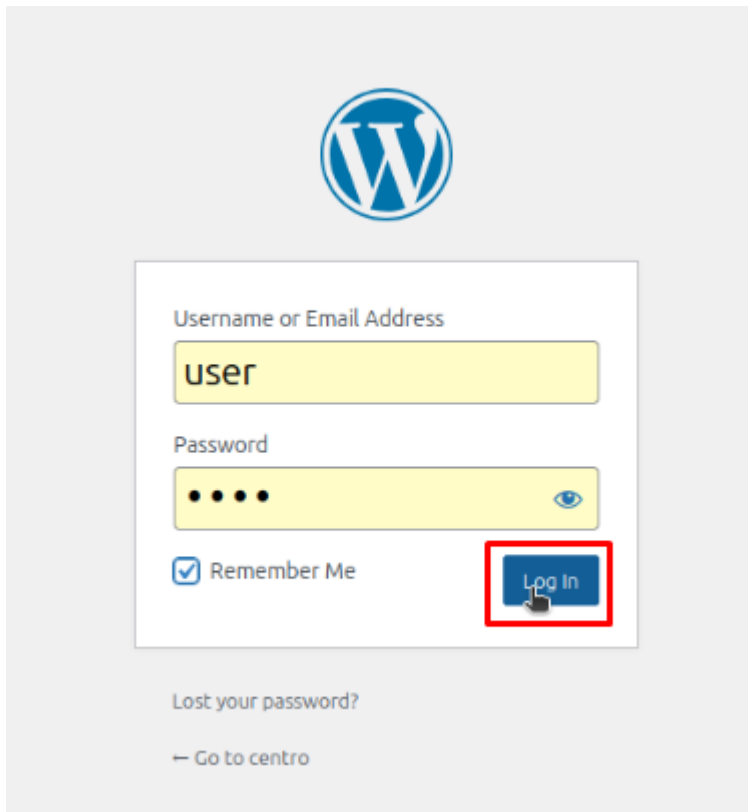
WordPress has been installed. Thank you, and enjoy!

**Username** user

**Password** *Your chosen password.*

[Log In](#)

Y hacemos LOG IN.



## -Activar el módulo “wsgi” para permitir la ejecución de aplicaciones Python

Introducimos los siguientes comandos:

```
sudo apt install libapache2-mod-wsgi-py3
```

```
sudo a2enmod wsgi
```

```
sudo systemctl reload apache2
```

```
user@user:~$ sudo apt install libapache2-mod-wsgi-py3 -y
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-wsgi-py3
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 18 no actualizados.
Se necesita descargar 103 kB de archivos.
Se utilizarán 300 kB de espacio de disco adicional después de esta operación.
Des:1 http://es.archive.ubuntu.com/ubuntu noble/main amd64 libapache2-mod-wsgi-py3 amd64 5.0.0-1build2 [103 kB]
Descargados 103 kB en 1s (180 kB/s)
Seleccionando el paquete libapache2-mod-wsgi-py3 previamente no seleccionado.
(Leyendo la base de datos ... 149668 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../libapache2-mod-wsgi-py3_5.0.0-1build2_amd64.deb ...
Desempaquetando libapache2-mod-wsgi-py3 (5.0.0-1build2) ...
Configurando libapache2-mod-wsgi-py3 (5.0.0-1build2) ...
apache2_invoke: Enable module wsgi
user@user:~$ sudo a2enmod wsgi
Module wsgi already enabled
user@user:~$ sudo systemctl reload apache2
user@user:~$
```

## -Crea y despliega una pequeña aplicación python para comprobar que funciona correctamente.

Nos dirigimos al directorio departamentos.centro.intranet con el comando: cd

var/www/departamentos.centro.intranet

Creamos un directorio app: mkdir app

Creamos un archivo dentro de app: sudo nano /app/app.wsgi

```
user@user:~$ cd /var/www/departamentos.centro.intranet
user@user:/var/www/departamentos.centro.intranet$ sudo mkdir app
[sudo] contraseña para user:
user@user:/var/www/departamentos.centro.intranet$ sudo nano /app/app.wsgi
user@user:/var/www/departamentos.centro.intranet$ sudo nano app/app.wsgi
user@user:/var/www/departamentos.centro.intranet$ cd app/
user@user:/var/www/departamentos.centro.intranet/app$ ls
app.wsgi
```

introducimos código python

```
GNU nano 7.2 /app/app.wsgi *
print("Hola mundo!!!!")

^G Ayuda ^O Guardar ^W Buscar ^K Cortar ^T Ejecutar ^C Ubicación
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar ^J Justificar ^_ Ir a línea
```

Tenemos que incluir wsgi, por lo cual nos dirigimos a sites-available usando el comando `cd /etc/apache2/sites-available/`

Modificamos el archivo `departamentos.centro.intranet` de la siguiente forma:

```
GNU nano 7.2 departamentos.centro.intranet.conf *
<VirtualHost *:80>
    ServerName departamentos.centro.intranet
    DocumentRoot /var/www/departamentos.centro.intranet

    WSGIScriptAlias / /var/www/departamentos.centro.intranet/app/app.wsgi

    <Directory /var/www/departamentos.centro.intranet/app>
        Require all granted
    </Directory>
</VirtualHost>
```

## -Adicionalmente protegeremos el acceso a la aplicación python mediante autenticación

Creamos un archivo `.htpasswd`

Tenemos que instalar una herramienta con el comando `sudo apt install apache2-utils`

Agregamos un usuario con el comando `sudo htpasswd -c /etc/apache2/.htpasswd admin`

```

user@user:/var/www/departamentos.centro.intranet$ sudo htpasswd -c /etc/apache2/.htpasswd
Usage:
    htpasswd [-cimBdpsDv] [-C cost] passwordfile username
    htpasswd -b[cmBdpsDv] [-C cost] passwordfile username password

    htpasswd -n[imBdps] [-C cost] username
    htpasswd -nb[mBdps] [-C cost] username password
-c Create a new file.
-n Don't update file; display results on stdout.
-b Use the password from the command line rather than prompting for it.
-i Read password from stdin without verification (for script usage).
-m Force MD5 encryption of the password (default).
-B Force bcrypt encryption of the password (very secure).
-C Set the computing time used for the bcrypt algorithm
  (higher is more secure but slower, default: 5, valid: 4 to 17).
-d Force CRYPT encryption of the password (8 chars max, insecure).
-s Force SHA encryption of the password (insecure).
-p Do not encrypt the password (plaintext, insecure).
-D Delete the specified user.
-v Verify password for the specified user.
On other systems than Windows and NetWare the '-p' flag will probably not work.
The SHA algorithm does not use a salt and is less secure than the MD5 algorithm.
user@user:/var/www/departamentos.centro.intranet$ sudo htpasswd -c /etc/apache2/.htpasswd user
New password:
Re-type new password:
Adding password for user user
user@user:/var/www/departamentos.centro.intranet$

```

Tenemos que cambiar la configuración usando sudo **nano** **/etc/apache2/sites-available/departamentos.centro.intranet.conf**  
Debe quedar así.

```

GNU nano 7.2 /etc/apache2/sites-available/departamentos.centro.intranet.conf *
<VirtualHost *:80>
    ServerName departamentos.centro.intranet
    DocumentRoot /var/www/departamentos.centro.intranet

    <Directory /var/www/departamentos.centro.intranet/app>
        AllowOverride All
        AuthType Basic
        AuthUserFile /etc/apache2/.htpasswd
        Require valid-user
    </Directory>
</VirtualHost>

```

Hacemos un reload de apache2 y listo

```

user@user:~$ sudo systemctl reload apache2
[sudo] contraseña para user:
user@user:~$

```

## -Instala y configura awstat.

Para instalar awstat necesitamos introducir el comando **sudo apt install awstats**  
Y tenemos que crear un archivo de configuración así que copiamos el predeterminado **sudo cp /etc/awstats/awstats.conf /etc/awstats/awstats.centro.intranet.conf**  
Abrimos el archivo e introducimos el dominio en estos sitios:

```
# Example: "domain.com"
#
SiteDomain="centro.intranet"

# Enter here all other possible domain names, addresses or virtual host
# aliases someone can use to access your site. Try to keep only the minimum
# number of possible names/addresses to have the best performances.
# You can repeat the "SiteDomain" value in this list.
# This parameter is used to analyze referer field in log file and to help
# AWStats to know if a referer URL is a local URL of same site or a URL of
# another site.
# Note: Use space between each value.
# Note: You can use regular expression values writing value with REGEX[value].
# Note: You can also use @/mypath/myfile if list of aliases are in a file.
# Example: "www.myserver.com localhost 127.0.0.1 REGEX[mydomain\.(net|org)$]"
#
HostAliases="localhost 127.0.0.1 centro.intranet"
```

Logformat lo cambiamos a 1

```
#
# Example for IIS:
# LogFormat = 2
#
LogFormat=1

# If your log field's separator is not a space, you can change this parameter.
# This parameter is not used if LogFormat is a predefined value (1,2,3,4)
```

Actualizamos con el comando `sudo /usr/lib/cgi-bin/awstats.pl -config=centro.intranet -update`

```
user@user:~$ sudo /usr/lib/cgi-bin/awstats.pl -config=centro.intranet -update
Check config file, permissions and Awstats documentation (in "/usr/lib/cgi-bin/awstats.pl" directory).
Create/Update database for config "/etc/awstats/awstats.centro.intranet.conf" by AWStats version 7.9 (build 20230108)
From data in log file "/var/log/apache2/access.log"...
Phase 1 : First bypass old records, searching new record...
Searching new records from beginning of log file...
Phase 2 : Now process new records (Flush history on disk after 20000 hosts)...
Reverse DNS lookup for ::1 not available without ipv6 plugin enabled.
Jumped lines in file: 0
Parsed lines in file: 132
Found 0 dropped records,
Found 0 comments,
Found 0 blank records,
Found 0 corrupted records,
Found 0 old records,
Found 132 new qualified records.
user@user:~$
```



-Instala un segundo servidor de tu elección (nginx, lighttpd) bajo el dominio “servidor2.centro.intranet”. Debes configurarlo para que sirva en el puerto 8080 y haz los cambios necesarios para ejecutar php. Instala phpmyadmin.

A la finalización del trabajo se procederá a una exposición de la presentación

Enlaces de interés

<https://uniwebsidad.com/libros/python/capitulo-13/python-bajo-apache>

Instrucciones de entrega

Se creará un repositorio en Github en el que se documentarán los pasos llevados a cabo para la realización de cada una de las actividades contempladas. La documentación debe incluir fragmentos de código empleados, además de imágenes que muestren la pantalla de la máquina virtual durante el proceso.

En enlace de dicho repositorio se incluirá en el repositorio del módulo, especificando claramente que se trata de la práctica de servidores web.

Fecha de entrega: La fecha límite de entrega será el 5 de diciembre.