

Práctica Servidores web 1º trimestre

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
- Activar los módulos necesarios para ejecutar php y acceder a mysql
- Instala y configura wordpress
- Activar el módulo “wsgi” para permitir la ejecución de aplicaciones Python
- Crea y despliega una pequeña aplicación python para comprobar que funciona correctamente.
- Adicionalmente protegeremos el acceso a la aplicación python mediante autenticación
- Instala y configura awstat.
- 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 documentará la instalación y configuración

Enviar a: jiprieto@ieslamarisma.net

Asunto: trabajo 2ASIR - <apellidos y nombre del alumno>

Adjuntar: Enlace al repositorio github en el que se expliquen los pasos y resultados obtenidos, además de adjuntar el contenido en formato pdf.

Fecha de entrega: La fecha límite de entrega será el 26 de noviembre.

Para la instalación del servidor apache tendremos que hacer lo siguiente.

1º Instalar Apache.

```

usuario@usuario-VirtualBox:~/Escritorio$ sudo apt update
[sudo] contraseña para usuario:
Des:1 http://security.ubuntu.com/ubuntu focal-security InRelease [
114 kB]
Obj:2 http://es.archive.ubuntu.com/ubuntu focal InRelease
Obj:3 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu focal-backports InReleas
e
Descargados 114 kB en 3s (36,3 kB/s)
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se pueden actualizar 113 paquetes. Ejecute «apt list --upgradable»
para verlos.

usuario@usuario-VirtualBox:~/Escritorio$ sudo apt install apache2
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.2-0
Paquetes sugeridos:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
Se instalarán los siguientes paquetes NUEVOS:
  apache2 apache2-bin apache2-data apache2-utils libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
  liblua5.2-0
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 113 no a
ctualizados.
Se necesita descargar 1.820 kB de archivos.
Se utilizarán 7.945 kB de espacio de disco adicional después de es
ta operación.

```

Nos preguntará si deseamos continuar con la instalación, le diremos sí.

```

¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libapr1
amd64 1.6.5-1ubuntu1 [91,4 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libapru
til1 amd64 1.6.1-4ubuntu2 [84,7 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libapru
til1-dbd-sqlite3 amd64 1.6.1-4ubuntu2 [10,5 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libapru
til1-ldap amd64 1.6.1-4ubuntu2 [8.736 B]
Des:5 http://es.archive.ubuntu.com/ubuntu focal/main amd64 liblua5
.2-0 amd64 5.2.4-1.1build3 [106 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64
apache2-bin amd64 2.4.41-4ubuntu3.8 [1.181 kB]
Des:7 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64
apache2-data all 2.4.41-4ubuntu3.8 [159 kB]
Des:8 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64
apache2-utils amd64 2.4.41-4ubuntu3.8 [84,5 kB]
Des:9 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64
apache2 amd64 2.4.41-4ubuntu3.8 [95,5 kB]
Descargados 1.820 kB en 1s (1.555 kB/s)
Seleccionando el paquete libapr1:amd64 previamente no seleccionado
.
(Leyendo la base de datos ... 85%

```

2º Para ver la lista de aplicaciones se mostrará de la siguiente manera.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo ufw app list
Aplicaciones disponibles:
  Apache
  Apache Full
  Apache Secure
  CUPS
```

- **Apache:** este perfil abre solo el puerto 80 (tráfico web normal no cifrado).
- **Apache Full:** este perfil abre los puertos 80 (tráfico web normal no cifrado) y 443 (tráfico TLS/SSL cifrado).
- **Apache Secure:** este perfil abre solo el puerto 443 (tráfico TLS/SSL cifrado).

3º Permitir a Apache utilizar únicamente el puerto 80.

Para ver de que el único puerto en uso de apache sea el 80 nos iremos a la carpeta **/etc/apache2** en el archivo **ports.conf**.

```
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Para la modificación del archivo hosts tendremos que dirigirnos hacia **/etc** y abrimos el archivo **hosts** con el siguiente comando.

```
usuario@usuario-VirtualBox:/$ cd etc
usuario@usuario-VirtualBox:/etc$ sudo nano hosts
```

GNU nano 4.8	hosts	Modificado
127.0.0.1	localhost	
127.0.1.1	usuario-VirtualBox	
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	

Para instalar php tendremos que introducir el siguiente comando.

```
usuario@usuario-VirtualBox:~$ sudo apt install php
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libapache2-mod-php7.4 php-common php7.4 php7.4-cli php7.4-common php7.4-json
  php7.4-opcache php7.4-readline
Paquetes sugeridos:
  php-pear
Se instalarán los siguientes paquetes NUEVOS:
```

Nos dirá si deseamos continuar con la instalación y pondremos "S".

```
?Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal/main amd64 php-common all 2:75 [
11,9 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4-common
amd64 7.4.3-4ubuntu2.7 [980 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4-json a
amd64 7.4.3-4ubuntu2.7 [19,2 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4-opcach
e amd64 7.4.3-4ubuntu2.7 [198 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4-readli
ne amd64 7.4.3-4ubuntu2.7 [12,6 kB]
```

Para acceder a mysql, tendremos que poner lo siguiente.

1º Instalarlo, donde tendremos que poner el siguiente comando y darle el permiso de que continúe con la instalación, poniéndole "S".

```
usuario@usuario-VirtualBox:~$ sudo apt install mysql-server
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libaio1 libcgi-fast-perl libcgi-pm-perl libevent-core-2.1-7
  libevent-pthreads-2.1-7 libfcgi-perl libhtml-template-perl libmecab2
  mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0
  mysql-client-core-8.0 mysql-server-8.0 mysql-server-core-8.0
Paquetes sugeridos:
  libipc-sharedcache-perl mailx tinycd
Se instalarán los siguientes paquetes NUEVOS:
  libaio1 libcgi-fast-perl libcgi-pm-perl libevent-core-2.1-7
  libevent-pthreads-2.1-7 libfcgi-perl libhtml-template-perl libmecab2
  mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0
  mysql-client-core-8.0 mysql-server mysql-server-8.0 mysql-server-core-8.0
0 actualizados, 16 nuevos se instalarán, 0 para eliminar y 3 no actualizados.
Se necesita descargar 31,5 MB de archivos.
Se utilizarán 262 MB de espacio de disco adicional después de esta operación.
```

```

¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-
core-8.0 amd64 8.0.27-0ubuntu0.20.04.1 [4.423 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-client-
8.0 amd64 8.0.27-0ubuntu0.20.04.1 [22,0 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libaio1 amd64 0.3.112
-5 [7.184 B]
Des:4 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libevent-core-2.1-7 a
md64 2.1.11-stable-1 [89,1 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libevent-pthreads-2.1
-7 amd64 2.1.11-stable-1 [7.372 B]
Des:6 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libmecab2 amd64 0.996
-10build1 [233 kB]
Des:7 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 mysql-server-
core-8.0 amd64 8.0.27-0ubuntu0.20.04.1 [18,4 MB]
22% [7 mysql-server-core-8.0 964 kB/18,4 MB 5%]

```

Instalaremos la librería con el siguiente comando, para poder comunicar mysql con php.

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

```

usuario@usuario-VirtualBox:~/Escritorio$ sudo apt-get install libapache2-mod-ph
p php-mysql
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  php7.4-mysql
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-php php-mysql php7.4-mysql
0 actualizados, 3 nuevos se instalarán, 0 para eliminar y 167 no actualizados.
Se necesita descargar 125 kB de archivos.
Se utilizarán 505 kB de espacio de disco adicional después de esta operación.

```

```

¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libapache2-mod-php a
ll 2:7.4+75 [2.836 B]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4-mysql
amd64 7.4.3-4ubuntu2.7 [121 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal/main amd64 php-mysql all 2:7.4+
75 [2.000 B]
Descargados 125 kB en 0s (307 kB/s)

```

2º Pondremos lo siguiente.

```
usuario@usuario-VirtualBox:~$ sudo mysql_secure_installation

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: y

There are three levels of password validation policy:

LOW      Length >= 8
MEDIUM  Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary
        file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 1
Please set the password for root here.

New password: 
```

Después de introducir la contraseña nos dirá si deseamos continuar con esa contraseña y podremos “Y” para confirmar que queremos continuar con esa contraseña. Más adelante, no hará más preguntas, donde responderemos a todo con “Y”.

3º Para acceder a MySQL, tendremos que introducir el siguiente comando.

```
usuario@usuario-VirtualBox:~$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.27-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2021, 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> 
```

Y para salirnos de MySQL, tan solo será necesario poner “exit”.

```
mysql> exit
Bye
usuario@usuario-VirtualBox:~$ 
```


Para instalar WordPress tendremos que hacer lo siguiente.

1º Crearemos una tabla con MySQL, con lo que antes que todo entraremos.

```
usuario@usuario-VirtualBox:/$ sudo mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 23
Server version: 8.0.27-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2021, 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>
```

2º Crearemos una tabla que se llamará centro.

CREATE DATABASE centro CHARACTER SET utf8 COLLATE utf8_general_ci;

```
mysql> CREATE DATABASE centro CHARACTER SET utf8 COLLATE utf8_general_ci;
Query OK, 1 row affected, 2 warnings (0,05 sec)

mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| centro   |
| information_schema |
| mydb     |
| mysql    |
| performance_schema |
| sys      |
+-----+
6 rows in set (0,00 sec)
```

3º Crearemos un usuario.

CREATE USER 'centrouuario'@'%' IDENTIFIED WITH mysql_native_password BY 'Centrouuario123_';

```
mysql> CREATE USER 'centrouuario'@'%' IDENTIFIED WITH mysql_native_password BY 'Centrouuario123_';
Query OK, 0 rows affected (0,01 sec)
```

4º Ahora haremos un acceso completo a la base de datos que se hará de la siguiente manera.

GRANT ALL ON centro.* TO 'centrouuario'@'%';

```
mysql> GRANT ALL ON centro.* TO 'centrouuario'@'%';
Query OK, 0 rows affected (0,01 sec)
```

5º Eliminaremos los privilegios.

```
mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0,01 sec)
```

6º Salimos MySQL.

```
mysql> exit
Bye
```

Ahora reiniciamos apache, que se puede hacer de las siguientes maneras.

```
usuario@usuario-VirtualBox:~$ sudo systemctl restart apache2
usuario@usuario-VirtualBox:~$ sudo service apache2 restart
```

Para ajustar la configuración de apache con el fin de permitir los reemplazos y reescritura en **.htaccess**.

1º Nos moveremos hacia **/etc/apache2/sites-available**.

```
usuario@usuario-VirtualBox:~$ cd /etc/apache2/sites-available
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ ls
000-default.conf  default-ssl.conf
```

2º Crearemos un fichero llamado **centro.conf** y lo configuraremos como se verá a continuación.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo nano centro.intranet.conf
```

```
<VirtualHost *:80>
    ServerName centro.intranet
    ServerAlias www.centro.intranet
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/centro.intranet
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

3º Habilitaremos el nuevo virtual host.

sudo a2ensite centro.intranet.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo a2ensite centro.intranet.conf
Enabling site centro.intranet.
To activate the new configuration, you need to run:
    systemctl reload apache2
```

4º Le hacemos un reload.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ systemctl reload apache2
usuario@usuario-VirtualBox:/etc/apache2/sites-available$
```

5º Deshabilitamos **000-default.conf**.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo a2dissite 000-default
Site 000-default disabled.
To activate the new configuration, you need to run:
    systemctl reload apache2
```

6º Haremos que la configuración esté segura.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK
usuario@usuario-VirtualBox:/etc/apache2/sites-available$
```

7º Volvemos a reiniciar apache.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo systemctl restart apache2
usuario@usuario-VirtualBox:/etc/apache2/sites-available$
```


Para configurar WordPress tendremos que hacer lo siguiente.

1º Habilitamos los reemplazos de **.htaccess** en **/etc/apache2/sites-available/centro.conf**.

```
usuario@usuario-VirtualBox:~$ cd /etc/apache2/sites-available
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo nano centro.intranet.conf
usuario@usuario-VirtualBox:/etc/apache2/sites-available$
```

```
<VirtualHost *:80>
    ServerName centro.intranet
    ServerAlias www.centro.intranet
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/centro.intranet
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    <Directory /var/www/centro.intranet/>
        AllowOverride All
    </Directory>
</VirtualHost>
```

2º Habilitaremos el módulo de reescritura.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
    systemctl restart apache2
```

3º Habilitamos los cambios.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo apache2ctl configtest
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message
Syntax OK
usuario@usuario-VirtualBox:/etc/apache2/sites-available$
```

4º Reiniciamos apache.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ systemctl restart apache2
usuario@usuario-VirtualBox:/etc/apache2/sites-available$
```

Para descargar WordPress haremos lo siguiente.

1º Nos movemos a la carpeta **tmp**.

2º Pero tendremos que instalar la orden **curl**.

3º Instalamos WordPress.

4º Tendremos que descomprimir el archivo, que nos hayamos descargado.

```
usuario@usuario-VirtualBox:~$ cd /tmp
usuario@usuario-VirtualBox:/tmp$ curl -O https://wordpress.org/latest.tar.gz

No se ha encontrado la orden «curl», pero se puede instalar con:

sudo apt install curl
```

```

usuario@usuario-VirtualBox:/tmp$ sudo apt install curl
[sudo] contraseña para usuario:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  curl
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 3 no actualizados.
Se necesita descargar 161 kB de archivos.
Se utilizarán 412 kB de espacio de disco adicional después de esta operación.
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 curl amd64 7.68.0-1ubuntu2.7 [161 kB]
Descargados 161 kB en 1s (227 kB/s)
Seleccionando el paquete curl previamente no seleccionado.
(Leyendo la base de datos ... 190348 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../curl_7.68.0-1ubuntu2.7_amd64.deb ...
Desempaquetando curl (7.68.0-1ubuntu2.7) ...
Configurando curl (7.68.0-1ubuntu2.7) ...
Procesando disparadores para man-db (2.9.1-1) ...

```

```

usuario@usuario-VirtualBox:/tmp$ curl -O https://wordpress.org/latest.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
 51 14.3M    51 7535k    0     0  1415k      0  0:00:10  0:00:05  0:00:05 1525k

```

```

usuario@usuario-VirtualBox:/tmp$ tar xzvf latest.tar.gz
wordpress/
wordpress/xmlrpc.php
wordpress/wp-blog-header.php
wordpress/readme.html
wordpress/wp-signup.php

```

5º Habrá que crear el siguiente archivo.

```

usuario@usuario-VirtualBox:/tmp/wordpress$ touch .htaccess
usuario@usuario-VirtualBox:/tmp/wordpress$

```

6º Copiaremos el archivo de configuración.

```

usuario@usuario-VirtualBox:/tmp/wordpress$ cp /tmp/wordpress/wp-config-sample.php /tmp/wordpress/wp-config.php
usuario@usuario-VirtualBox:/tmp/wordpress$

```

7º Crearemos un directorio para las actualizaciones.

```

usuario@usuario-VirtualBox:/tmp/wordpress$ mkdir /tmp/wordpress/wp-content/upgrade
usuario@usuario-VirtualBox:/tmp/wordpress$

```

8º Copiamos todo el contenido de la carpeta **WordPress** hacia la carpeta del dominio **/var/www/centro** e incluiremos los archivos que están ocultos.

```

usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo cp -a /tmp/wordpress/. /var/www/centro.intranet
usuario@usuario-VirtualBox:/etc/apache2/sites-available$

```

Por último, comprobaremos que todos los archivos se han copiado correctamente.

```

usuario@usuario-VirtualBox:/var/www/centro.intranet$ ls
index.php  readme.html  wp-admin  wp-comments-post.php  wp-config-sample.php  wp-cron.php  wp-links-opml.php  wp-login.php  wp-settings.php  wp-trackback.php
license.txt  wp-activate.php  wp-blog-header.php  wp-config.php  wp-content  wp-includes  wp-load.php  wp-mail.php  wp-signup.php  xmlrpc.php

```

Para realizar las configuraciones del directorio de WordPress tendremos que hacer lo siguiente. Actualizar la propiedad con el siguiente comando, con el fin de permitir modificar la propiedad del archivo.

```

usuario@usuario-VirtualBox:/etc/apache2/sites-available$ cd ~
usuario@usuario-VirtualBox:~$ sudo chown -R www-data:www-data /var/www/centro.intranet
[sudo] contraseña para usuario:
usuario@usuario-VirtualBox:~$

```

A continuación, ejecutaremos dos comandos con **find** para establecer los permisos correctamente de los directorios y archivos de WordPress.

```
usuario@usuario-VirtualBox:~$ ls /var/www
centro.intranet html
usuario@usuario-VirtualBox:~$ sudo find /var/www/centro.intranet/ -type d -exec chmod 750 {} \;
[sudo] contraseña para usuario:
usuario@usuario-VirtualBox:~$ sudo find /var/www/centro.intranet/ -type f -exec chmod 640 {} \;
usuario@usuario-VirtualBox:~$
```

Estos permisos deberán de hacer que pueda trabajar de forma efectiva con WordPress, pero hay que tener en cuenta que algunos complementos y procedimientos para que puedan requerir ajustes adicionales.

Ahora tendremos que generar las claves secretas de WordPress.

```
usuario@usuario-VirtualBox:~$ curl -s https://api.wordpress.org/secret-key/1.1/salt/
define('AUTH_KEY', 'S%[{Z2~qd+_bwiL%(tx^|(oz+_mbkYFlavb;wUtHEM]pRq}o0gx02@Dc6!BU/sw+');
define('SECURE_AUTH_KEY', ');T4 [yCJnXhLMoA|`=Z,+>H-DcbQj(g.{ci~L>4RF+YRe+6DA%0Y nt0[Y..LYb');
define('LOGGED_IN_KEY', '0sZwHPo,e`0))^bkZ{!W$/Q;`*7kL@!%5Pd?)mnYPww|D>FVZ`k,Z9xivH?}7Yf.');
define('NONCE_KEY', '+yEQFR~T+!/g)$>H.wc<90o|8_m?Vr~$JVBts}Qbe|d0--qXJC(68au-H@WJrvr)');
define('AUTH_SALT', '-0-+lNpYDQgtC]D0D#966 P}vxrN:0*g(_oh]W+_pXD%o|/iVi&9{Ba0NJ>Hm+>@');
define('SECURE_AUTH_SALT', 'v5oCZ>>8Z+A^7_LR@Wi-mry[J^058q-j#PB$M&%?25!]fb*1}M|e}pF*qR{kmVH');
define('LOGGED_IN_SALT', '7G3.nLp-JA#LdB-p!]J;01Z_A]Dcf:GzR|MfU@1~ ~,$E#+f*m|HuNFmni[mZ9qj');
define('NONCE_SALT', '+iem, 0*F+!aF6k$4=J}xQ})0J*%-Ij3M1b@.>t*Qqe@|,StSvzFRr`KJvInK+^V');
usuario@usuario-VirtualBox:~$
```

Las claves secretas, las copiamos en el siguiente archivo dentro del directorio **centro.intranet**.

```
usuario@usuario-VirtualBox:~$ sudo nano /var/www/centro.intranet/wp-config.php
usuario@usuario-VirtualBox:~$
```

Dentro del archivo **wp-config.php** tendremos que configurar lo siguiente.

```
define( 'DB_NAME', 'centro' );

/** MySQL database username */
define( 'DB_USER', 'centrouuario' );

/** MySQL database password */
define( 'DB_PASSWORD', 'Centrouuario123_' );

/** MySQL hostname */
define( 'DB_HOST', 'localhost' );

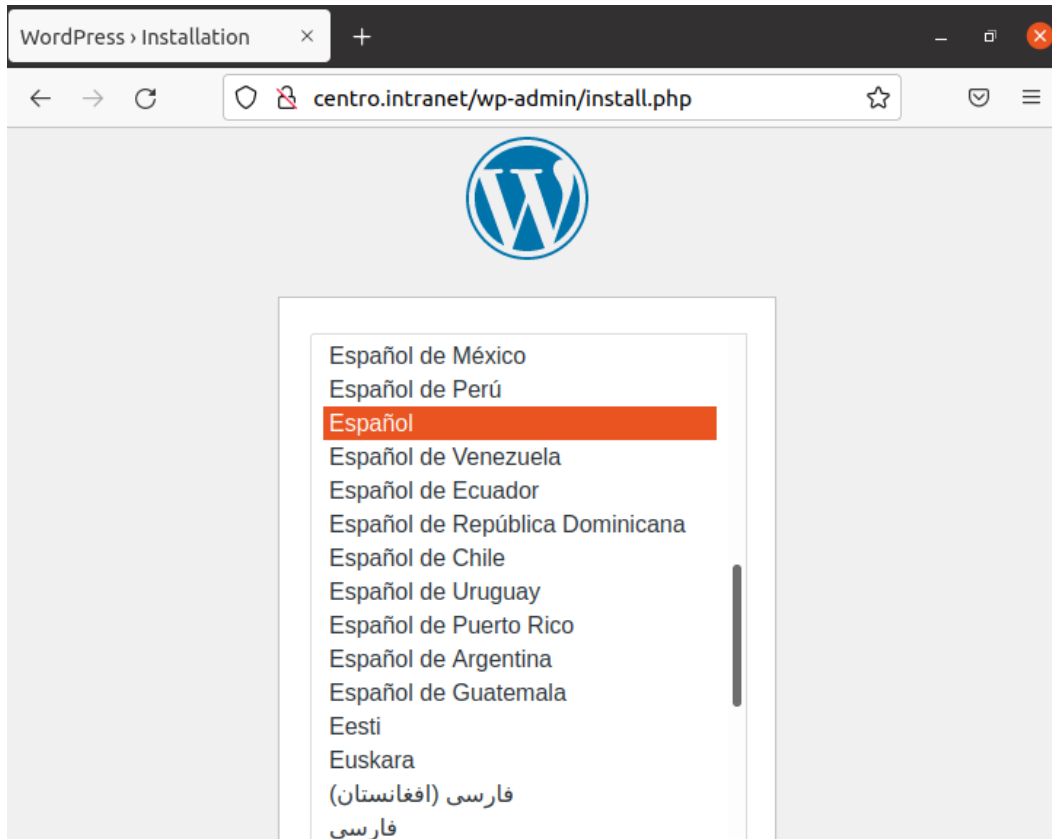
/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );

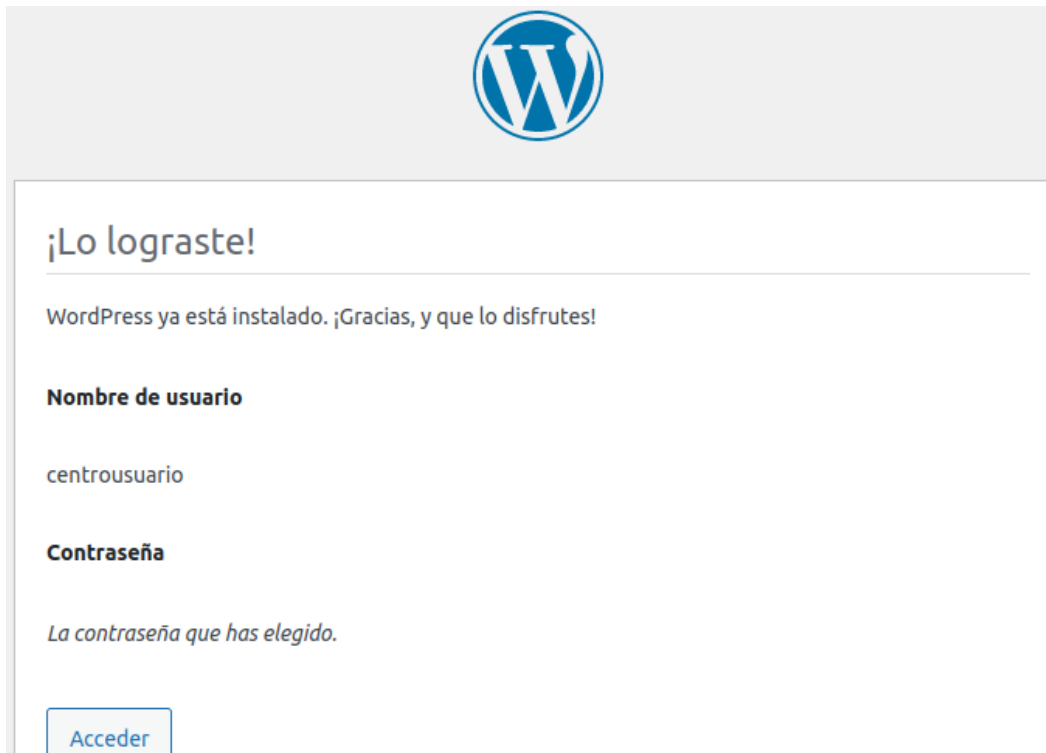
/**#@+
 * Authentication unique keys and salts.
 *
 * Change these to different unique phrases! You can generate these using
 * the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service}.
 *
 * You can change these at any point in time to invalidate all existing cookies.
 * This will force all users to have to log in again.
 *
 * @since 2.6.0
 */
define('AUTH_KEY', 'S%[{Z2~qd+_bwiL%(tx^|(oz+_mbkYFlavb;wUtHEM]pRq}o0gx02@Dc6!BU/sw+');
define('SECURE_AUTH_KEY', ');T4 [yCJnXhLMoA|`=Z,+>H-DcbQj(g.{ci~L>4RF+YRe+6DA%0Y nt0[Y..LYb');
define('LOGGED_IN_KEY', '0sZwHPo,e`0))^bkZ{!W$/Q;`*7kL@!%5Pd?)mnYPww|D>FVZ`k,Z9xivH?}7Yf.');
define('NONCE_KEY', '+yEQFR~T+!/g)$>H.wc<90o|8_m?Vr~$JVBts}Qbe|d0--qXJC(68au-H@WJrvr)');
define('AUTH_SALT', '-0-+lNpYDQgtC]D0D#966 P}vxrN:0*g(_oh]W+_pXD%o|/iVi&9{Ba0NJ>Hm+>@');
define('SECURE_AUTH_SALT', 'v5oCZ>>8Z+A^7_LR@Wi-mry[J^058q-j#PB$M&%?25!]fb*1}M|e}pF*qR{kmVH');
define('LOGGED_IN_SALT', '7G3.nLp-JA#LdB-p!]J;01Z_A]Dcf:GzR|MfU@1~ ~,$E#+f*m|HuNFmni[mZ9qj');
define('NONCE_SALT', '+iem, 0*F+!aF6k$4=J}xQ})0J*%-Ij3M1b@.>t*Qqe@|,StSvzFRr`KJvInK+^V');
```

```
/* Add any custom values between this line and the "stop editing" line. */  
define('FS_METHOD', 'direct');  
/* That's all, stop editing! Happy publishing. */
```

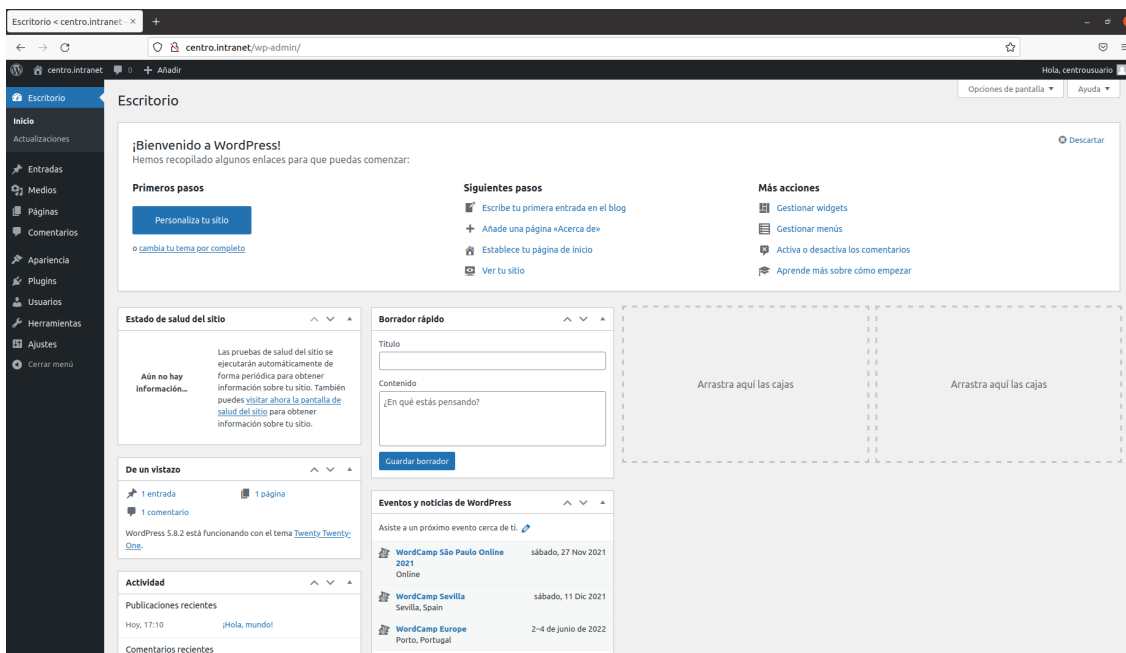
Después de todo lo anterior, comprobamos si funciona la conexión con WordPress desde el navegador.



Para terminar con la configuración de WordPress tendremos que crearnos un usuario con el nombre de usuario, contraseña y correo electrónico.



The image shows the WordPress installation success screen. At the top is the WordPress logo. Below it, the text reads: "¡Lo lograste!" followed by "WordPress ya está instalado. ¡Gracias, y que lo disfrutes!". There are two input fields: "Nombre de usuario" with the value "centrouuario" and "Contraseña" with the placeholder text "La contraseña que has elegido.". At the bottom is a blue button labeled "Acceder".



The image shows the WordPress dashboard. The top bar includes the site name "centro.intranet" and the user "Hola, centrouuario". The left sidebar contains a menu with options like "Inicio", "Actualizaciones", "Entradas", "Medios", "Páginas", "Comentarios", "Apariencia", "Plugins", "Usuarios", "Herramientas", "Ajustes", and "Cerrar menú". The main content area is titled "Escritorio" and includes a welcome message "¡Bienvenido a WordPress!". It features several widgets: "Primeros pasos" with links to "Personaliza tu sitio" and "cambia tu tema por completo"; "Sigüientes pasos" with tasks like "Escribe tu primera entrada en el blog"; "Más acciones" with options like "Gestionar widgets"; "Estado de salud del sitio" showing site health; "Borrador rápido" for creating new posts; "De un vistazo" showing site statistics; "Actividad" showing recent publications; and "Eventos y noticias de WordPress" listing upcoming events like "WordCamp São Paulo Online 2021".

A continuación, habilitaremos los módulos necesarios para python, configuraciones e instalaciones que la requieran.

1º Habilitamos el módulo wsgi.

```
usuario@usuario-VirtualBox:~$ sudo apt-get install libapache2-mod-wsgi
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libpython2.7-stdlib libpython2.7-minimal libpython2.7-stdlib python2 python2-minimal python2.7 python2.7-minimal
Paquetes sugeridos:
  python2-doc python2-tk python2.7-doc binutils binfmt-support
Se instalarán los siguientes paquetes NUEVOS:
  libapache2-mod-wsgi libpython2.7-stdlib libpython2.7 libpython2.7-minimal libpython2.7-stdlib python2 python2-minimal python2.7 python2.7-minimal
0 actualizados, 9 nuevos se instalarán, 0 para eliminar y 166 no actualizados.
Se necesita descargar 4.945 kB de archivos.
Se utilizarán 20,3 MB de espacio de disco adicional después de esta operación.

¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/universe amd64 libpython2.7-minimal amd64 2.7.18-1-20.04.1 [335 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/universe amd64 python2.7-minimal amd64 2.7.18-1-20.04.1 [1.285 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 python2-minimal amd64 2.7.17-2ubuntu4 [27,5 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu focal-updates/universe amd64 libpython2.7-stdlib amd64 2.7.18-1-20.04.1 [1.887 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu focal-updates/universe amd64 python2.7 amd64 2.7.18-1-20.04.1 [248 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 libpython2-stdlib amd64 2.7.17-2ubuntu4 [7.072 B]
Des:7 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 python2 amd64 2.7.17-2ubuntu4 [26,5 kB]
Des:8 http://es.archive.ubuntu.com/ubuntu focal-updates/universe amd64 libpython2.7 amd64 2.7.18-1-20.04.1 [1.038 kB]
Des:9 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 libapache2-mod-wsgi amd64 4.6.8-1ubuntu3 [90,8 kB]
Descargados 4.945 kB en 4s (1.193 kB/s)
```

2º Crearemos una estructura de directorios para una aplicación.

En primer lugar, tendremos que crear un directorio donde vamos a montar toda la aplicación.

```
usuario@usuario-VirtualBox:~$ mkdir /home/usuario/departamentos-python
usuario@usuario-VirtualBox:~$ mkdir /home/usuario/departamentos-python/trunk
usuario@usuario-VirtualBox:~$ mkdir /home/usuario/departamentos-python/trunk/departamentos-web
usuario@usuario-VirtualBox:~$
```

Dentro de este directorio, dividiremos su arquitectura en dos partes.

La que se encarga de almacenar nuestra aplicación python (**departamentoapp**) y la que se encarga de servir a la aplicación donde almacenamos archivos estáticos (**departamento_html**).

```
usuario@usuario-VirtualBox:~/departamentos-python/trunk/departamentos-web$ mkdir departamentoapp
usuario@usuario-VirtualBox:~/departamentos-python/trunk/departamentos-web$ mkdir departamento_html
usuario@usuario-VirtualBox:~/departamentos-python/trunk/departamentos-web$
```

Además, crearemos una carpeta donde se almacenarán los logs de errores y accesos a nuestra Web App.

```
usuario@usuario-VirtualBox:~/departamentos-python/trunk/departamentos-web$ mkdir logs
usuario@usuario-VirtualBox:~/departamentos-python/trunk/departamentos-web$
```

3º Para crear un controlador para la aplicación, es decir, todas las peticiones realizadas por los usuarios y serán manejadas por un único archivo, que estará almacenado en nuestro directorio **departamentoapp**.

```
usuario@usuario-VirtualBox:~/departamentos-python/trunk/departamentos-web$ echo '# -*- coding: utf-8 -*-' > departamentoapp/controller.py
usuario@usuario-VirtualBox:~/departamentos-python/trunk/departamentos-web$
```


El archivo **controller.py** actuará como un pseudo, con el fin de encargarse de manejar todas las peticiones del usuario, haciendo la llamada a los módulos correspondientes según las URI.

El módulo tan solo se encargará de definir una función, que actúe con cada petición del usuario.

Esta función será una función WSGI aplicación válida, con lo que:

1º Tendrá que llamarse **aplicacion**.

2º Tendrá que recibir dos parámetros: **environ**, del módulo **os**, que provee un diccionario de las peticiones HTTP estándar y otras variables de entorno, y la función **start_response**, de WSGI, encargada de entregar la respuesta HTTP al usuario.

```
# -*- coding: utf-8 -*-

def application(environ, start_response):
    # Genero la salida HTML a mostrar al usuario
    output = "<p>Bienvenido a mi <b>PythonApp</b>!!!</p>"
    # Inicio una respuesta al navegador
    start_response('200 OK', [('Content-Type', 'text/html; charset=utf-8')])
    # Retorno el contenido HTML
    return output
```

3º Para configurar el VirtualHost, en ella se encontrará la clave para correr nuestra aplicación Python a través de la Web.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo nano departamentos.centro.intranet.conf
usuario@usuario-VirtualBox:/etc/apache2/sites-available$

<VirtualHost *:80>
    ServerName departamentos.centro.intranet
    DocumentRoot /var/www/departamentos-python/trunk/departamentos-web/departamento_html
    WSGIScriptAlias / /var/www/departamentos-python/trunk/departamentos-web/departamentoapp/controller.py
    ErrorLog /var/www/departamentos-python/trunk/departamentos-web/logs/errors.log
    CustomLog /var/www/departamentos-python/trunk/departamentos-web/logs/access.log combined

    <Directory />
        Options FollowSymLinks
        AllowOverride All
    </Directory>
</VirtualHost>
```

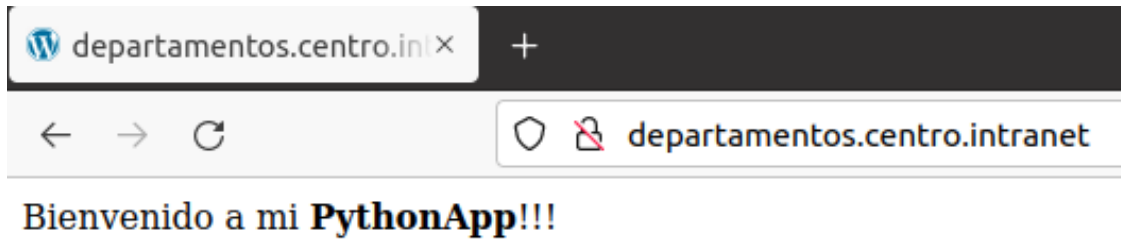
Ahora habilitaremos el sitio web.

```
usuario@usuario-VirtualBox:/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
```

Recargamos apache.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ systemctl reload apache2
usuario@usuario-VirtualBox:/etc/apache2/sites-available$
```

Habilitamos el sitio del host.



Para la protección de acceso a la aplicación de python mediante una autenticación añadiremos lo siguiente en `/etc/apache2/apache2.conf`, lo que vendría siendo el último `<Directory>`.

```
# Sets the default security model of the Apache2 HTTPD server. It does
# not allow access to the root filesystem outside of /usr/share and /var/www.
# The former is used by web applications packaged in Debian,
# the latter may be used for local directories served by the web server. If
# your system is serving content from a sub-directory in /srv you must allow
# access here, or in any related virtual host.
<Directory />
    Options FollowSymLinks
    AllowOverride None
    Require all denied
</Directory>

<Directory /usr/share>
    AllowOverride None
    Require all granted
</Directory>

<Directory /var/www/>
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>

<Directory "/var/www/departamentos-python">
    AuthType Basic
    AuthName "Restricted Files"
    AuthBasicProvider file
    AuthUserFile "/usr/local/apache/passwd/passwords"
    Require user usuario
</Directory>
```

Y haremos un `systemctl apache2 restart`.

Una vez que lo hayamos añadido, abriremos el navegador y pondremos el enlace del dominio, donde nos saldrá lo siguiente.

 departamentos.centro.intranet

This site is asking you to sign in.

Username

Password

Cancel

Sign in

Para la instalación del paquete AWStats pondremos lo siguiente.

```
usuario@usuario-VirtualBox:/etc/apache2/sites-available$ sudo apt-get install awstats
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libnet-xwhois-perl
Paquetes sugeridos:
  libnet-dns-perl libnet-ip-perl libgeo-ipfree-perl
Se instalarán los siguientes paquetes NUEVOS:
  awstats libnet-xwhois-perl
0 actualizados, 2 nuevos se instalarán, 0 para eliminar y 166 no actualizados.
Se necesita descargar 1.861 kB de archivos.
Se utilizarán 7.057 kB de espacio de disco adicional después de esta operación.

¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 awstats all 7.6+dfsg-2ubuntu0.20.04.1 [1.840 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libnet-xwhois-perl all 0.90-4 [21,0 kB]
Descargados 1.861 kB en 2s (1.063 kB/s)
```

Después de la instalación tendremos que habilitar el módulo CGI.

```
usuario@usuario-VirtualBox:~$ sudo a2enmod cgi
Enabling module cgi.
To activate the new configuration, you need to run:
  systemctl restart apache2
usuario@usuario-VirtualBox:~$
```

Y reiniciamos apache.

```
usuario@usuario-VirtualBox:~$ systemctl restart apache2
usuario@usuario-VirtualBox:~$
```

Para configurar AWStats tendremos que hacer lo siguiente.

```
usuario@usuario-VirtualBox:~$ sudo cp /etc/awstats/awstats.conf /etc/awstats/awstats.test.com.conf
usuario@usuario-VirtualBox:~$
```

Tendremos que abrir el archivo que hemos hecho la copia e introduciremos lo siguiente.

```

usuario@usuario-VirtualBox:/etc/awstats$ ls
awstats.centro.intranet.conf  awstats.conf  awstats.conf.local
usuario@usuario-VirtualBox:/etc/awstats$ sudo nano awstats.centro.intranet.conf
usuario@usuario-VirtualBox:/etc/awstats$

# "LogFile" contains the web, ftp or mail server log file to analyze.
# Possible values: A full path, or a relative path from awstats.pl directory.
# Example: "/var/log/apache/access.log"
# Example: "../logs/mycombinedlog.log"
# You can also use tags in this filename if you need a dynamic file name
# depending on date or time (Replacement is made by AWStats at the beginning
# of its execution). This is available tags :
#   %YYYY-n   is replaced with 4 digits year we were n hours ago
#   %YY-n     is replaced with 2 digits year we were n hours ago
#   %MM-n     is replaced with 2 digits month we were n hours ago
#   %MO-n     is replaced with 3 letters month we were n hours ago
#   %DD-n     is replaced with day we were n hours ago
#   %HH-n     is replaced with hour we were n hours ago
#   %NS-n     is replaced with number of seconds at 00:00 since 1970
#   %WM-n     is replaced with the week number in month (1-5)
#   %Wm-n     is replaced with the week number in month (0-4)
#   %WY-n     is replaced with the week number in year (01-52)
#   %Wy-n     is replaced with the week number in year (00-51)
#   %DW-n     is replaced with the day number in week (1-7, 1=sunday)
#               use n=24 if you need (1-7, 1=monday)
#   %Dw-n     is replaced with the day number in week (0-6, 0=sunday)
#               use n=24 if you need (0-6, 0=monday)
#   Use 0 for n if you need current year, month, day, hour...
# Example: "/var/log/access_log.%YYYY-0%MM-0%DD-0.log"
# Example: "C:/WINNT/system32/LogFiles/W3SVC1/ex%YY-24%MM-24%DD-24.log"
# You can also use a pipe if log file come from a pipe :
# Example: "gzip -cd /var/log/apache/access.log.gz |"
# If there are several log files from load balancing servers :
# Example: "/pathtotools/logresolvemerge.pl *.log |"
#
LogFile="/var/log/apache2/access.log"

```

```
# "SiteDomain" must contain the main domain name, or the main intranet web
# server name, used to reach the web site.
# If you share the same log file for several virtual web servers, this
# parameter is used to tell AWStats to filter record that contains records for
# this virtual host name only (So check that this virtual hostname can be
# found in your log file and use a personalized log format that include the
# %virtualname tag).
# But for multi hosting a better solution is to have one log file for each
# virtual web server. In this case, this parameter is only used to generate
# full URL's links when ShowLinksOnUrl option is set to 1.
# If analyzing mail log, enter here the domain name of mail server.
# Example: "myintranetserver"
# Example: "www.domain.com"
# Example: "ftp.domain.com"
# 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 an 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="www.centro.intranet localhost 127.0.0.1"

# When this parameter is set to 1, AWStats adds a button on report page to
# allow to "update" statistics from a web browser. Warning, when "update" is
# made from a browser, AWStats is run as a CGI by the web server user defined
# in your web server (user "nobody" by default with Apache, "IUSR_XXX" with
# IIS), so the "DirData" directory and all already existing history files
# awstatsMMYYYY[.xxx].txt must be writable by this user. Change permissions if
# necessary to "Read/Write" (and "Modify" for Windows NTFS file systems).
# Warning: Update process can be long so you might experience "time out"
# browser errors if you don't launch AWStats frequently enough.
# When set to 0, update is only made when AWStats is run from the command
# line interface (or a task scheduler).
# Possible values: 0 or 1
# Default: 0
#
AllowToUpdateStatsFromBrowser=1
```

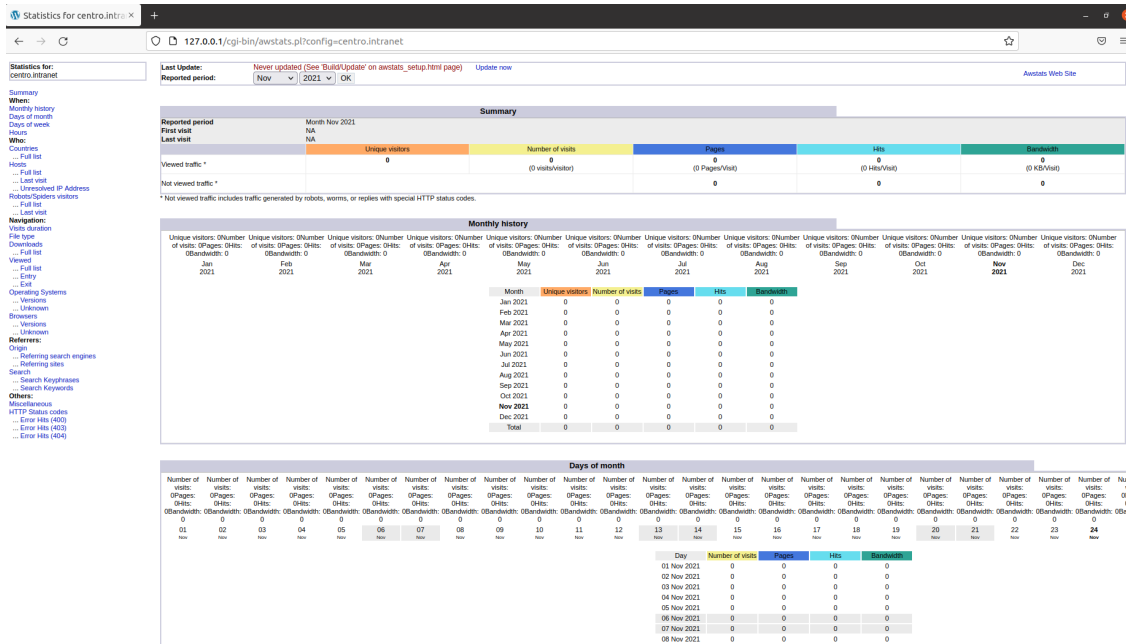
Después de haber hecho los cambios anteriores, tendremos que construir unas estadísticas iniciales que se harán a partir de los registros actuales en el servidor, y lo haremos con el siguiente comando.

```
usuario@usuario-VirtualBox:~$ sudo /usr/lib/cgi-bin/awstats.pl -config=test.com -update
Create/Update database for config "/etc/awstats/awstats.test.com.conf" by AWStats version 7.6 (build 20161204)
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)...
Jumped lines in file: 0
Parsed lines in file: 124
Found 0 dropped records,
Found 0 comments,
Found 0 blank records,
Found 0 corrupted records,
Found 0 old records,
Found 124 new qualified records.
```

Para hacer la configuración de Apache para AWStats tendremos que poner los siguientes comandos.

```
usuario@usuario-VirtualBox:~$ sudo cp -r /usr/lib/cgi-bin /var/www/html/
usuario@usuario-VirtualBox:~$ sudo chown www-data:www-data /var/www/html/cgi-bin/
usuario@usuario-VirtualBox:~$ sudo chmod -R 755 /var/www/html/cgi-bin/
usuario@usuario-VirtualBox:~$
```

Para terminar visitaremos AWStats por medio de la siguiente url.



Para actualizar los registros haremos lo siguiente.

```
usuario@usuario-VirtualBox:/etc/awstats$ sudo nano /etc/crontab
usuario@usuario-VirtualBox:/etc/awstats$
```

Introducimos la última línea.

```
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# Example of job definition:
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .----- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | | |
# * * * * * user-name command to be executed

17 * * * * root cd / && run-parts --report /etc/cron.hourly
25 6 * * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * root test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
#
*/10 * * * * root /usr/lib/cgi-bin/awstats.pl -config=centro.intranet -update
```


Tendremos que instalar Nginx con el siguiente comando.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo apt install nginx
[sudo] contraseña para usuario:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter
  libnginx-mod-mail libnginx-mod-stream nginx-common nginx-core
Paquetes sugeridos:
  fcgiwrap nginx-doc
Se instalarán los siguientes paquetes NUEVOS:
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter
  libnginx-mod-mail libnginx-mod-stream nginx nginx-common nginx-core
0 actualizados, 7 nuevos se instalarán, 0 para eliminar y 165 no actualizados.
Se necesita descargar 603 kB de archivos.
Se utilizarán 2.134 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 nginx-common
all 1.18.0-0ubuntu1.2 [37,5 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libnginx-mod-
http-image-filter amd64 1.18.0-0ubuntu1.2 [14,4 kB]
Des:3 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libnginx-mod-
http-xslt-filter amd64 1.18.0-0ubuntu1.2 [12,7 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libnginx-mod-
mail amd64 1.18.0-0ubuntu1.2 [42,5 kB]
Des:5 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libnginx-mod-
stream amd64 1.18.0-0ubuntu1.2 [67,3 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 nginx-core am
d64 1.18.0-0ubuntu1.2 [425 kB]
68% [6 nginx-core 230 kB/425 kB 54%]
```

Y visualizamos la lista de aplicaciones.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo ufw app list
Aplicaciones disponibles:
  Apache
  Apache Full
  Apache Secure
  CUPS
  Nginx Full
  Nginx HTTP
  Nginx HTTPS
```

Habilitaremos el que sea menos restrictivo, es decir, a Nginx HTTP con el siguiente comando.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo ufw allow 'Nginx HTTP'
Reglas actualizadas
Reglas actualizadas (v6)
```

También habilitaremos el cortafuegos y comprobaremos su estado.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo ufw enable
El cortafuegos está activo y habilitado en el arranque del sistema
usuario@usuario-VirtualBox:~/Escritorio$ sudo ufw status
Estado: activo

Hasta          Acción          Desde
-----
Nginx HTTP     ALLOW           Anywhere
Nginx HTTP (v6) ALLOW           Anywhere (v6)
```

Antes de iniciar Nginx tendremos que apagar Apache e iniciaremos Nginx.

```
usuario@usuario-VirtualBox:~/Escritorio$ systemctl stop apache2

usuario@usuario-VirtualBox:~/Escritorio$ systemctl start nginx
usuario@usuario-VirtualBox:~/Escritorio$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2021-11-25 17:35:49 CET; 2s ago
     Docs: man:nginx(8)
   Process: 3897 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 3898 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Main PID: 3899 (nginx)
    Tasks: 2 (limit: 3419)
   Memory: 3.3M
   CGroup: /system.slice/nginx.service
           └─3899 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             └─3900 nginx: worker process

nov 25 17:35:48 usuario-VirtualBox systemd[1]: Starting A high performance web server and a reverse proxy server...
nov 25 17:35:49 usuario-VirtualBox systemd[1]: Started A high performance web server and a reverse proxy server.
```

A continuación, tendremos que crear un dominio para Nginx.

Le asignamos propietario del directorio y se le dará los siguientes permisos.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo mkdir -p /var/www/servidor2.centro.intranet/html
usuario@usuario-VirtualBox:~/Escritorio$ sudo chown -R $USER:$USER /var/www/servidor2.centro.intranet/html
usuario@usuario-VirtualBox:~/Escritorio$ sudo chmod -R 755 /var/www/servidor2.centro.intranet
usuario@usuario-VirtualBox:~/Escritorio$
```

Crearemos un archivo html dentro del dominio.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo nano /var/www/servidor2.centro.intranet/html/index.html
usuario@usuario-VirtualBox:~/Escritorio$
```

```
<html>
  <head>
    <title>BIENVENIDO AL DOMINIO DE NGINX</title>
  </head>
  <body>
    <h1>ESTE ES EL DOMINIO SERVIDOR2.CENTRO.INTRANET</h1>
  </body>
</html>
```

A continuación, crearemos un virtual host para el dominio de Nginx.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo nano /etc/nginx/sites-available/servidor2.centro.intranet
usuario@usuario-VirtualBox:~/Escritorio$
```

```
server {
    listen 8080;
    listen [::]:8080;

    root /var/www/servidor2.centro.intranet/html;
    index index.html index.htm index.nginx-debian.html;

    server_name servidor2.centro.intranet www.servidor2.centro.intranet;

    location / {
        try_files $uri $uri/ =404;
    }
}
```

Ahora habilitaremos el archivo (`/etc/nginx/sites-available/servidor2.centro.intranet`) creando un enlace con el directorio **sites-enabled**.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo ln -s /etc/nginx/sites-available/servidor2.centro.intranet /etc/nginx/sites-enabled/
usuario@usuario-VirtualBox:~/Escritorio$
```

Para que no se produzca ningún error abriremos y modificaremos el siguiente archivo.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo nano /etc/nginx/nginx.conf
usuario@usuario-VirtualBox:~/Escritorio$
```

```
server_names_hash_bucket_size 64;
# server_name_in_redirect off;
```

Ahora vamos a comprobar que no haya ningún error de sintaxis con el siguiente comando.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

Reiniciamos Nginx y añadimos el dominio en el archivo **etc/hosts**.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo systemctl restart nginx
usuario@usuario-VirtualBox:~/Escritorio$ sudo nano /etc/hosts
```

```
127.0.0.1    localhost
127.0.1.1    usuario-VirtualBox
127.0.0.1    centro.intranet
127.0.0.1    departamentos.centro.intranet
127.0.0.1    servidor2.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
```

Y comprobamos que funcione entrando desde el navegador con la siguiente ruta.



Continuaremos configurando el virtual host añadiendo las siguientes líneas de **location**.

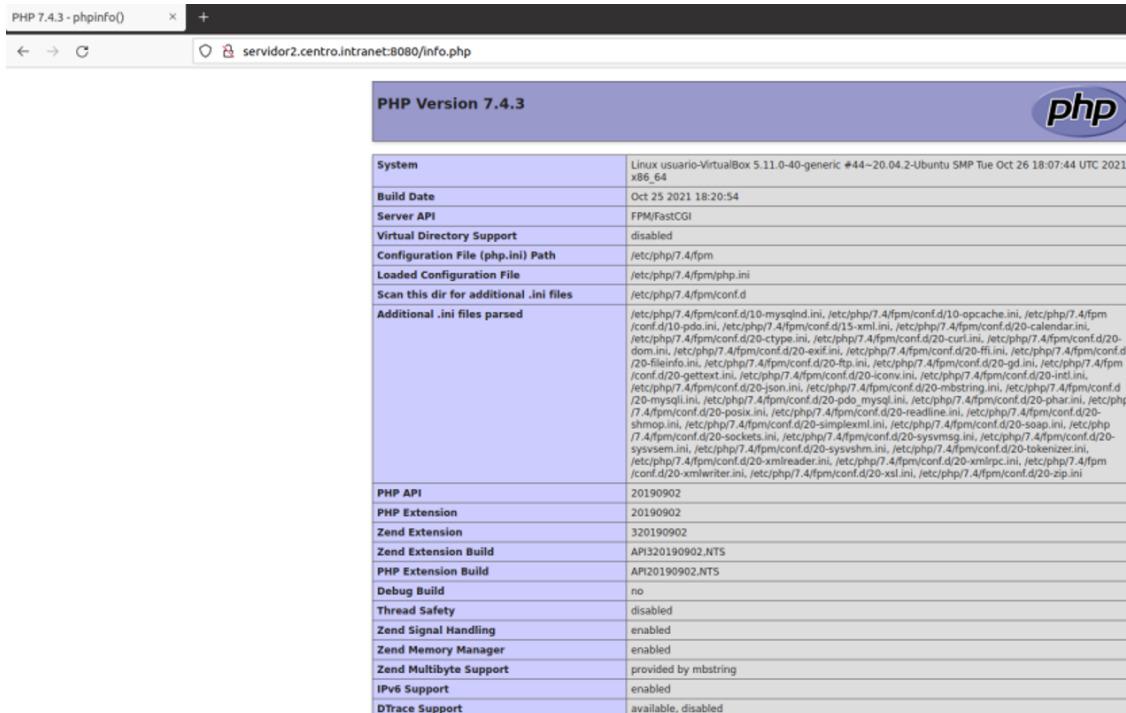
```
server {  
    listen 8080;  
    listen [::]:8080;  
  
    root /var/www/servidor2.centro.intranet/html;  
    index index.html index.htm index.nginx-debian.html;  
  
    server_name servidor2.centro.intranet www.servidor2.centro.intranet;  
  
    location / {  
        try_files $uri $uri/ =404;  
    }  
  
    location ~ \.php$ {  
        include snippets/fastcgi-php.conf;  
        fastcgi_pass unix:/var/run/php/php7.4-fpm.sock;  
    }  
  
    location ~ /\.ht {  
        deny all;  
    }  
}
```

Haremos una configuración del **info.php**.

```
usuario@usuario-VirtualBox:~/Escritorio$ sudo nano /var/www/servidor2.centro.intranet/info.php  
usuario@usuario-VirtualBox:~/Escritorio$
```

```
<?php  
phpinfo();
```

Comprobamos que el servidor php funcione entrando desde el navegador.



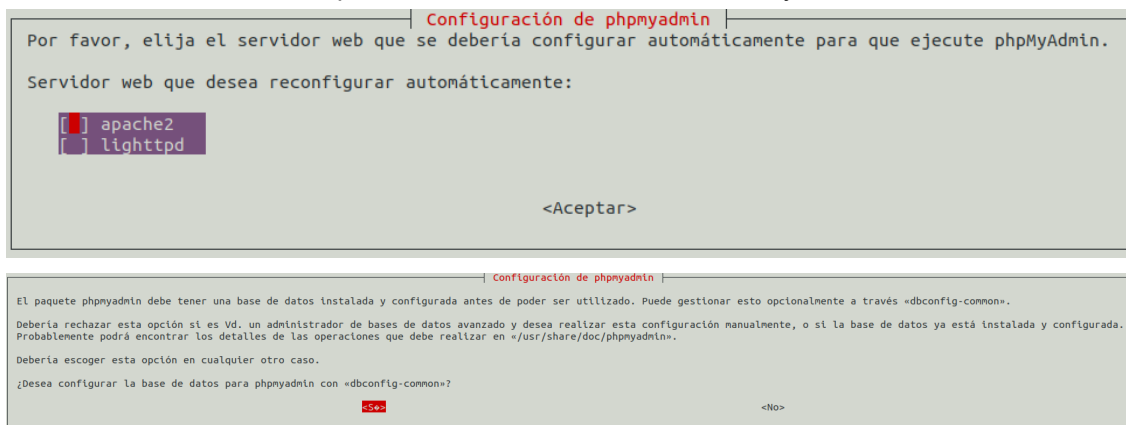
PHP Version 7.4.3	
System	Linux usuario-VirtualBox 5.11.0-40-generic #44~20.04.2-Ubuntu SMP Tue Oct 26 18:07:44 UTC 2021 x86_64
Build Date	Oct 25 2021 18:20:54
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/fpm
Loaded Configuration File	/etc/php/7.4/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/fpm/conf.d
Additional .ini files parsed	/etc/php/7.4/fpm/conf.d/10-mysqld.ini, /etc/php/7.4/fpm/conf.d/10-opcache.ini, /etc/php/7.4/fpm/conf.d/10-pdo.ini, /etc/php/7.4/fpm/conf.d/15-xml.ini, /etc/php/7.4/fpm/conf.d/20-calendar.ini, /etc/php/7.4/fpm/conf.d/20-ctype.ini, /etc/php/7.4/fpm/conf.d/20-curl.ini, /etc/php/7.4/fpm/conf.d/20-dom.ini, /etc/php/7.4/fpm/conf.d/20-exif.ini, /etc/php/7.4/fpm/conf.d/20-ffi.ini, /etc/php/7.4/fpm/conf.d/20-fileinfo.ini, /etc/php/7.4/fpm/conf.d/20-ftp.ini, /etc/php/7.4/fpm/conf.d/20-gd.ini, /etc/php/7.4/fpm/conf.d/20-gettext.ini, /etc/php/7.4/fpm/conf.d/20-iconv.ini, /etc/php/7.4/fpm/conf.d/20-intl.ini, /etc/php/7.4/fpm/conf.d/20-json.ini, /etc/php/7.4/fpm/conf.d/20-mbstring.ini, /etc/php/7.4/fpm/conf.d/20-mysqli.ini, /etc/php/7.4/fpm/conf.d/20-pdo_mysql.ini, /etc/php/7.4/fpm/conf.d/20-phar.ini, /etc/php/7.4/fpm/conf.d/20-posix.ini, /etc/php/7.4/fpm/conf.d/20-readline.ini, /etc/php/7.4/fpm/conf.d/20-shmop.ini, /etc/php/7.4/fpm/conf.d/20-simplexml.ini, /etc/php/7.4/fpm/conf.d/20-soap.ini, /etc/php/7.4/fpm/conf.d/20-sockets.ini, /etc/php/7.4/fpm/conf.d/20-sysmsg.ini, /etc/php/7.4/fpm/conf.d/20-syssem.ini, /etc/php/7.4/fpm/conf.d/20-sysvshm.ini, /etc/php/7.4/fpm/conf.d/20-tokenizer.ini, /etc/php/7.4/fpm/conf.d/20-xmlreader.ini, /etc/php/7.4/fpm/conf.d/20-xmlrpc.ini, /etc/php/7.4/fpm/conf.d/20-xmlwriter.ini, /etc/php/7.4/fpm/conf.d/20-xsl.ini, /etc/php/7.4/fpm/conf.d/20-zip.ini
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902.NTS
PHP Extension Build	API20190902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	available, disabled

Después de la instalación de php, instalaremos phpMyAdmin con el siguiente comando.

```
usuario@usuario-VirtualBox:~/www/servidor2.centro.intranet$ sudo apt install phpmyadmin
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
dbconfig-common dbconfig-mysql gcc-profiles-free javascript-common libjs-jquery libjs-openlayers libjs-sphinxdoc libjs-underscore libnig5 libzip5 php-bz2 php-curl php-gd php-google-recaptcha
php-mbstring php-phpmyadmin-metatrator php-phpmyadmin-shapefile php-phpmyadmin-sql-parser php-phoselib php-psr-cache php-psr-container php-psr-log php-symfony-cache php-symfony-cache-contracts
php-symfony-expression-language php-symfony-service-contracts php-symfony-var-exporter php-tcpdf php-twig php-twig-extensions php-xml php-zip php7.4-bz2 php7.4-curl php7.4-gd php7.4-mbstring
php7.4-xml php7.4-zip
Paquetes sugeridos:
php-dbase php-libzodium php-mcrypt php-gmp php-symfony-service-implementation php-inagick php-twig-doc php-symfony-translation php-recode php-gd2 php-pragmarx-google2fa php-bacon-qrcode
php-sanyoul-u2f-php-server
Paquetes recomendados:
php-mcrypt
Se instalarán los siguientes paquetes NUEVOS:
dbconfig-common dbconfig-mysql gcc-profiles-free javascript-common libjs-jquery libjs-openlayers libjs-sphinxdoc libjs-underscore libnig5 libzip5 php-bz2 php-curl php-gd php-google-recaptcha
php-mbstring php-phpmyadmin-metatrator php-phpmyadmin-shapefile php-phpmyadmin-sql-parser php-phoselib php-psr-cache php-psr-container php-psr-log php-symfony-cache php-symfony-cache-contracts
php-symfony-expression-language php-symfony-service-contracts php-symfony-var-exporter php-tcpdf php-twig php-twig-extensions php-xml php-zip php7.4-bz2 php7.4-curl php7.4-gd php7.4-mbstring
php7.4-xml php7.4-zip phpmyadmin
0 actualizados, 39 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
Se necesita descargar 15,9 MB de archivos.
Se utilizarán 71,3 MB de espacio de disco adicional después de esta operación.

¿Desea continuar? [S/n] s
Des:1 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 dbconfig-common all 2.0.13 [586 kB]
Des:2 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 dbconfig-mysql all 2.0.13 [1.136 B]
Des:3 http://es.archive.ubuntu.com/ubuntu focal/main amd64 gcc-profiles-free all 2.0.14-dfsg-1 [549 kB]
Des:4 http://es.archive.ubuntu.com/ubuntu focal/main amd64 javascript-common all 11 [6.066 B]
Des:5 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libjs-jquery all 3.3.1-dfsg-3 [329 kB]
Des:6 http://es.archive.ubuntu.com/ubuntu focal-updates/main amd64 libjs-underscore all 1.9.1-dfsg-1ubuntu0.20.04.1 [99,5 kB]
Des:7 http://es.archive.ubuntu.com/ubuntu focal/main amd64 libjs-sphinxdoc all 1.8.5-7ubuntu3 [97,1 kB]
Des:8 http://es.archive.ubuntu.com/ubuntu focal/universe amd64 libnig5 amd64 6.9.4-1 [142 kB]
12% [8 libnig5 0 B/142 kB 0%]
```

Seleccionamos el servidor Apache2, continuará con la instalación y le daremos sí.



Configuración de phpmyadmin

Por favor, elija el servidor web que se debería configurar automáticamente para que ejecute phpMyAdmin.

Servidor web que desea reconfigurar automáticamente:

☒ apache2

☐ lighttpd

<Aceptar>

<No>

El paquete phpmyadmin debe tener una base de datos instalada y configurada antes de poder ser utilizado. Puede gestionar esto opcionalmente a través «dbconfig-common».

Debería rechazar esta opción si es Vd. un administrador de bases de datos avanzado y desea realizar esta configuración manualmente, o si la base de datos ya está instalada y configurada. Probablemente podrá encontrar los detalles de las operaciones que debe realizar en «/usr/share/doc/phpmyadmin».

Debería escoger esta opción en cualquier otro caso.

¿Desea configurar la base de datos para phpmyadmin con «dbconfig-common»? ☒ Sí ☐ No

Tendremos que introducir una contraseña y le daremos **Aceptar**.

Nos pedirá una confirmación de la contraseña y le daremos **Aceptar**.

Y terminará con la instalación.

Nos saldrá un error donde tendremos que darle a la opción interrumpir/abortar.

Para solucionar esto tendremos que meternos en **mysql** y escribiremos lo siguiente.

```
usuario@usuario-VirtualBox:/var/www/servidor2.centro.intranet$ sudo mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 19
Server version: 8.0.27-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2021, 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> UNINSTALL COMPONENT "file://component_validate_password";
Query OK, 0 rows affected (2,58 sec)
```

Volvemos a intentar instalar **phpmyadmin**.

```
usuario@usuario-VirtualBox:/var/www/servidor2.centro.intranet$ sudo apt install phpmyadmin
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
phpmyadmin ya está en su versión más reciente (4:4.9.5+dfsg1-2).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
```

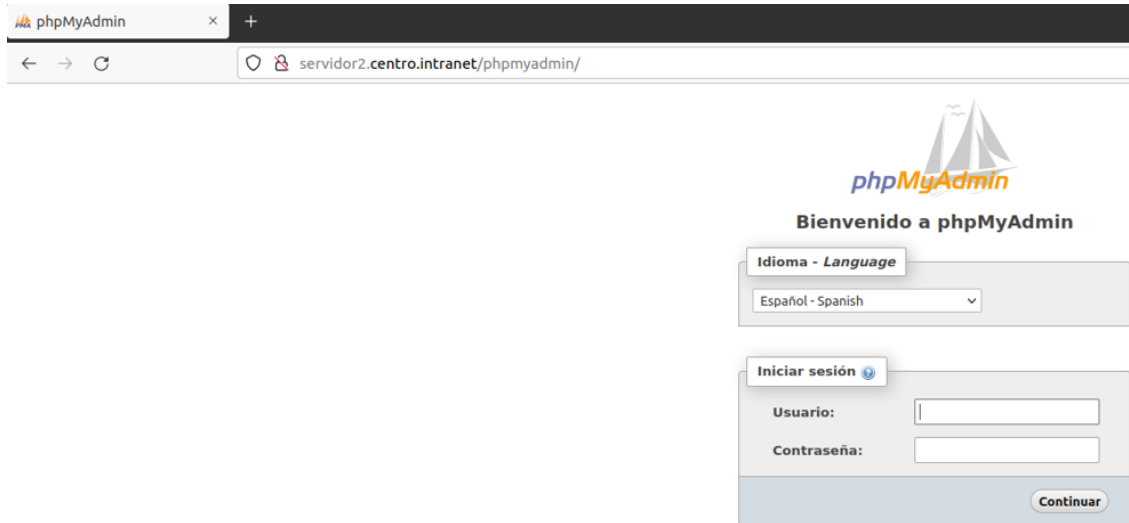
Volvemos a entrar en **mysql** y pondremos lo siguiente.

```
mysql> INSTALL COMPONENT "file://component_validate_password";
Query OK, 0 rows affected (0,01 sec)
```

Crearemos un enlace simbólico del siguiente archivo.

```
usuario@usuario-VirtualBox:/var/www/servidor2.centro.intranet$ sudo ln -s /usr/share/phpmyadmin /var/www/servidor2.centro.intranet/phpmyadmin
usuario@usuario-VirtualBox:/var/www/servidor2.centro.intranet$
```


Una vez que lo hayamos hecho, tendremos que reiniciar y buscaremos en el navegador.



The screenshot shows a web browser window with a single tab titled 'phpMyAdmin'. The address bar displays the URL 'servidor2.centro.intranet/phpmyadmin/'. The page content features the phpMyAdmin logo at the top, followed by the text 'Bienvenido a phpMyAdmin'. Below this, there is a language selection section labeled 'Idioma - Language' with a dropdown menu currently set to 'Español - Spanish'. Further down is the login section, labeled 'Iniciar sesión', which contains two input fields: 'Usuario:' and 'Contraseña:'. A 'Continuar' button is located at the bottom right of the login form.