

SERVIDOR WEB

Hoy en día casi todos los centros educativos cuentan con una página web donde poder ofrecer información del centro como las enseñanzas que se imparten, horarios escolares, ubicación del mismo, fotos y un largo etc de cosas relativas al mundo educativo y del propio centro. Además en muchas ocasiones podemos ver como en un centro nos vendría muy bien tener una plataforma donde los profesores y alumnos tuvieran un punto en común donde poder subir y descargar materiales, ponerse en contacto entre ellos, o saber cuando es la fecha de un examen por ejemplo.

La ventaja de tener el servidor web alojado en el propio centro es que no dependemos de un servicio de **hosting**, pagado en la mayoría de los casos, para administrar nuestra página y probar cosas. Podemos probar todo lo que queramos y tener una gestión directa de todo lo que acontece al servidor. Otro punto positivo, es que las peticiones que se hagan desde dentro de la red del centro, irán bastante más rápidas comparándolas cuando estas se hacen a un servidor externo.

Como contrapunto, el equipo que haga de servidor debe estar siempre encendido si queremos que los alumnos y profesores puedan acceder a su información en el momento que quieran. Además la potencia del ordenador servidor y el ancho de banda de la conexión de nuestro centro, delimitará las posibilidades del mismo cuando tenga que atender a varias peticiones simultáneas de información.

Nosotros vamos a aprovechar que en esta unidad vamos a aprender a montar un Servidor Web en Linux, para además instalarle una plataforma de software libre como es Moodle.

A partir de aquí empezaremos instalando los recursos necesarios para que nuestro servidor aloje una página web con acceso a bases de datos SQL y código PHP.

LAMP

LAMP son las siglas de **Linux + Apache + MySQL + PHP/Perl/Python** y en resumen es un paquete con todo lo necesario para poder instalar un servidor web con la mayoría de las funcionalidades que vamos a necesitar.

Quitando Linux por ser obvio, una breve descripción de los elementos anteriores:

- Apache:** servidor web HTTP de código abierto, potente y muy estable usado en la mayoría de los servidores web de Internet.

El archivo principal de configuración de Apache es **/etc/apache2/apache2.conf**

- MySQL:** Sistema de gestión de Bases de Datos que usa el modelo relacional con SQL.

El archivo principal de configuración de MySQL es **/etc/mysql/my.cnf**

- PHP:** lenguaje de programación diseñado para obtener páginas web dinámicas.

El archivo principal de configuración de PHP es **/etc/php5/apache2/php.ini**

Más información:

<http://es.wikipedia.org/wiki/LAMP>

INSTALACIÓN

Seguramente si hemos seguido los pasos ya tendremos instalado LAMP en nuestro sistema servidor pero de no ser así lo instalaremos usando el comando:

sudo taskset install lamp-server

Probamos que el servicio MySQL está activo con

service mysql status

Y vemos como entre los puerto a la escucha de nuestro servidor deben estar el 80 para el servidor web (HTTP) y el 3306 del servicio de base de datos (MySQL).

netstat -putan |more

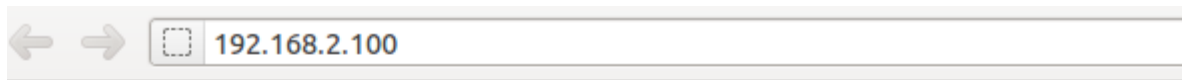
Conexiones activas de Internet (servidores y establecidos)

Proto	Recib	Enviad	Dirección local	Dirección remota	Estado
PID/Program name					
tcp	0	0	127.0.0.1:3306	0.0.0.0:*	ESCUCHAR
tcp	0	0	0.0.0.0:139	0.0.0.0:*	ESCUCHAR
tcp	0	0	0.0.0.0:110	0.0.0.0:*	ESCUCHAR
tcp	0	0	0.0.0.0:143	0.0.0.0:*	ESCUCHAR
tcp	0	0	0.0.0.0:80	0.0.0.0:*	ESCUCHAR
tcp	0	0	0.0.0.0:2323	0.0.0.0:*	ESCUCHAR
tcp	0	0	0.0.0.0:25	0.0.0.0:*	ESCUCHAR
tcp	0	0	0.0.0.0:445	0.0.0.0:*	ESCUCHAR
tcp	0	0	0.0.0.0:993	0.0.0.0:*	ESCUCHAR
tcp	0	0	0.0.0.0:995	0.0.0.0:*	ESCUCHAR
tcp6	0	0	:::110	:::*	ESCUCHAR
tcp6	0	0	:::143	:::*	ESCUCHAR
tcp6	0	0	:::2323	:::*	ESCUCHAR

PROBANDO NUESTRO SERVIDOR

Hay muchas formas para comprobar que el servidor web Apache está activo, pero tal vez la mejor sea probarlo desde algún ordenador cliente intentando cargar la página de inicio por defecto que Apache lleva una vez instalado. Para eso abrimos un **navegador web** en cualquier ordenador cliente de la red, o en el equipo anfitrión en el caso de que el servidor este virtualizado y cargamos la dirección **IP del servidor**. El resultado debe ser el contenido de la página **index.html** situada en el directorio **/var/www**

En el caso de que quisiéramos probar directamente en el propio servidor si nuestra página está activa, como no disponemos de ningún navegador web ya que a nuestro sistema no le hemos instalado entorno gráfico nos iremos a la carpeta **/var/www** y una vez allí conectaremos mediante **telnet** usando



It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

N

telnet 127.0.1.1 80

Una vez hemos conectado cargaremos la página con el comando

GET /index.html

El resultado debe mostrar el contenido de la página

```
enrique@ServidorCep:/var$ cd www
enrique@ServidorCep:/var/www$ ls
index.html
enrique@ServidorCep:/var/www$ telnet 127.0.1.1 80
Trying 127.0.1.1...
Connected to 127.0.1.1.
Escape character is '^I'.
GET /index.html
<html><body><h1>It works!</h1>
<p>This is the default web page for this server.</p>
<p>The web server software is running but no content has been added, yet.</p>
</body></html>
Connection closed by foreign host.
enrique@ServidorCep:/var/www$
```

Vamos a probar por último que nuestro servidor puede **ejecutar código PHP** sin problemas. Para ello nos dirigimos a la carpeta var/www y creamos un archivo de texto usando alguno de los editores de texto como por ejemplo nano

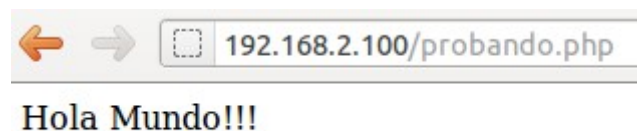
sudo nano probando.php

Modificamos el contenido añadiéndole estas líneas, guardamos y cerramos.

<?php echo 'Hola Mundo!!!'; ?>

```
GNU nano 2.2.6 Archivo: probando.php
<?php
echo 'Hola Mundo!!!';
?>
```

A continuación nos vamos a un navegador de algún cliente en red y cargamos la dirección **IP del servidor /la_página.php** que acabamos de crear para ver el resultado.



Todo funciona correcto, el servidor web, las bases de datos MySQL y PHP ejecuta correctamente el código.

Para poder acceder externamente a las bases de datos de nuestro servidor es necesario modificar el archivo de configuración de MySQL para ello nos iremos al archivo de configuración del mismo **/etc/mysql/my.cnf** y lo editaremos para que apunte a la IP de nuestro servidor.

`sudo nano /etc/mysql/my.cnf`

y modificamos la línea donde permite sólo acceder desde el **localhost** por nuestra IP del servidor.

`bind-adress = 192.168.2.100`

Guardamos y salimos.

```
#
user            = mysql
socket          = /var/run/mysqld/mysqld.sock
port            = 3306
basedir         = /usr
datadir         = /var/lib/mysql
tmpdir          = /tmp
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address    = 192.168.2.100
#
# * Fine Tuning
#
key_buffer      = 16M
max_allowed_packet = 16M
thread_stack    = 192K
thread_cache_size = 8
# This replaces the startup script and checks MyISAM tables if needed
# the first time they are touched
myisam-recover  = BACKUP
#max_connections = 100
#table_cache    = 64
```

Reiniciaremos el servicio para que tengan efecto los cambios

sudo service mysql restart

```
enrique@ServidorCep:/var/www$ sudo service mysql restart
[sudo] password for enrique:
mysql start/running, process 1213
enrique@ServidorCep:/var/www$
```

Como por motivos de seguridad **no es recomendable usar el superusuario root para acceder desde otros equipos** vamos a crearnos un nuevo usuario con permisos de administrador que usaremos para administrar las bases de datos. Para crearlo usamos el comando

mysql -u root -p

Ahora es cuando tenemos que recurrir a la memoria, o mejor aún a ese papel o archivo donde guardamos las claves y usuarios que usamos durante la instalación de nuestro servidor, para poder acceder como **root a MySQL**.

Una vez dentro del **SQL Monitor** usaremos un comando para crearnos un usuario que en mi caso le he llamado **enriquesql** con pass “0000” y luego le daremos todos los privilegios como administrador.

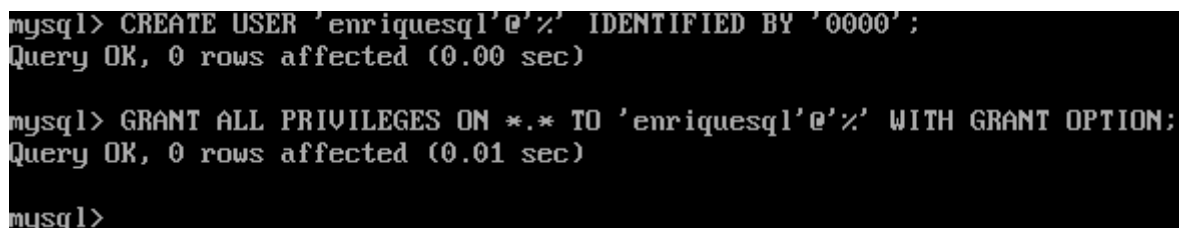
Para crear al usuario usamos

```
CREATE USER 'nombre_usuario'@'%' IDENTIFIED BY 'password';
```

```
CREATE USER 'enriquesql'@'%' IDENTIFIED BY '0000';
```

Y para darle todos los permisos usamos el comando

```
GRANT ALL PRIVILEGES ON *.* TO 'nombre_usuario'@'%' WITH GRANT OPTION;
```



```
mysql> CREATE USER 'enriquesql'@'%' IDENTIFIED BY '0000';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'enriquesql'@'%' WITH GRANT OPTION;
Query OK, 0 rows affected (0.01 sec)

mysql>
```

Una vez realizados los cambios salimos con

```
exit
```

PHPMyAdmin

Una vez tenemos todo lo necesario en nuestro servidor vamos a aprovecharnos de una herramienta que nos va a facilitar mucho la interacción con las bases de datos de MySQL, esta herramienta es **PHPMyAdmin**.

PHPMyAdmin es una aplicación escrita usando código PHP que nos va a permitir **manejar las bases de datos a través de una página web con un interfaz gráfico** ahorrándonos muchos de los **comandos SQL** que tendríamos que usar hasta para las tareas más básicas.

Más información:

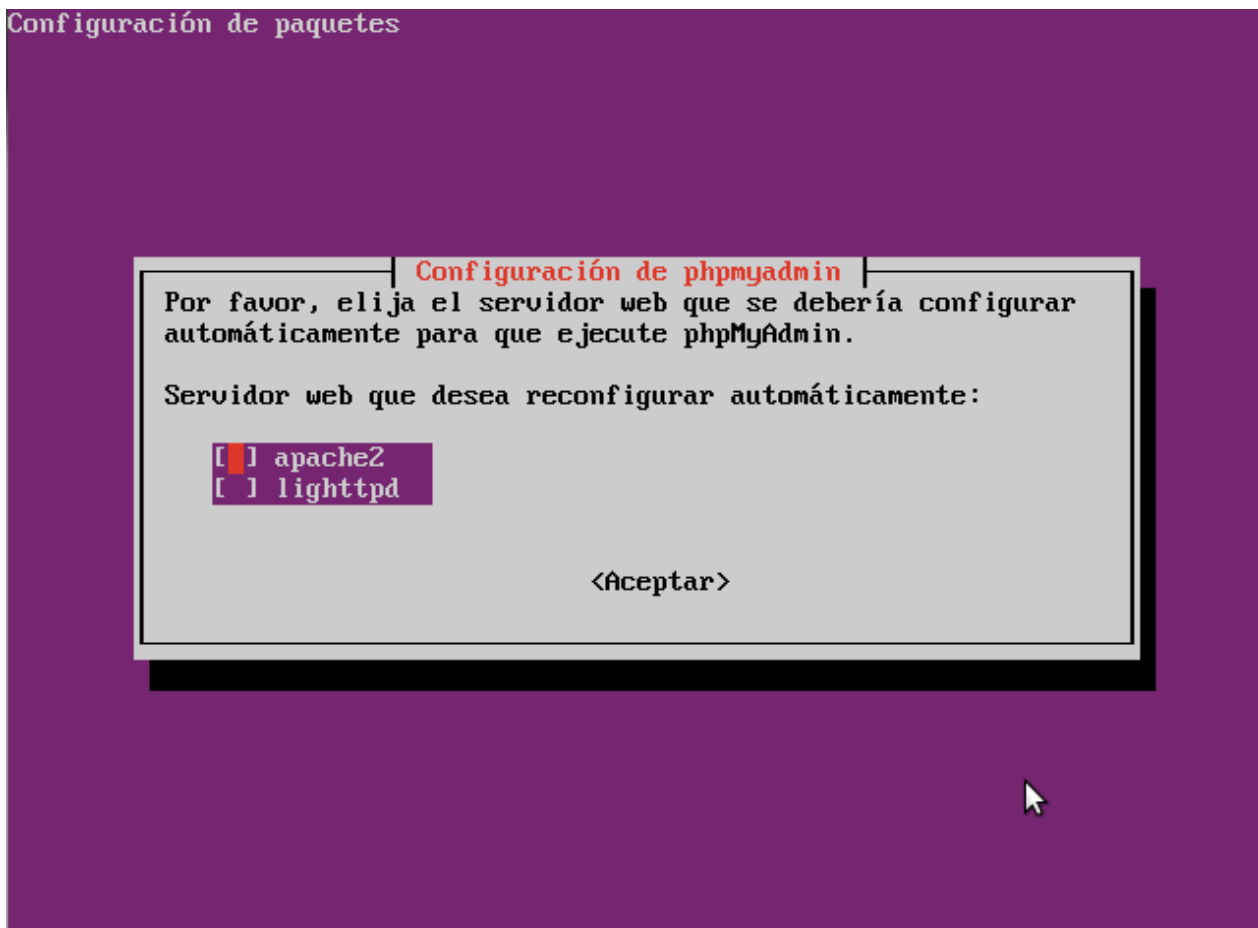
<http://es.wikipedia.org/wiki/PhpMyAdmin>

http://www.phpmyadmin.net/home_page/index.php

INSTALACIÓN

Para iniciar la instalación de PHPMyAdmin usaremos **apt-get** en una terminal mediante el comando

```
sudo apt-get install phpmyadmin
```



Elegimos “**apache2**” y **aceptamos**. Nos avisará que para empezar a usar PHPMyAdmin necesitamos tener una base de datos creada y configurada, para simplificar seleccionamos “**Sí**”.

Para confirmar nos va a pedir el **password de root de MySQL**, la introducimos.

Necesitamos además, una clave que nos permita usar PHPMyAdmin y conectarlo con las bases de datos de MySQL. Podemos usar la misma clave que tenemos para MySQL o otra diferente. En nuestro caso, a pesar de ser menos seguro que usar una nueva clave, **usaremos la misma** para no tener que estar recordando o apuntando más claves.

Configuración de paquetes

Configuración de phpmyadmin

Es necesario tener una base de datos instalada y configurada para phpmyadmin antes de poder utilizarlo. Puede gestionar esto opcionalmente a través «dbconfig-common».

Si ud. es un administrador de bases de datos avanzado o si la base de datos ya está instalada y configurada quizás quiera realizar esta configuración manualmente, y debería rechazar esta opción. 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>

Aunque siempre es posible usar una nueva clave si así lo deseamos.

Repetimos la clave y aceptamos.

PROBANDO PHPMyAdmin

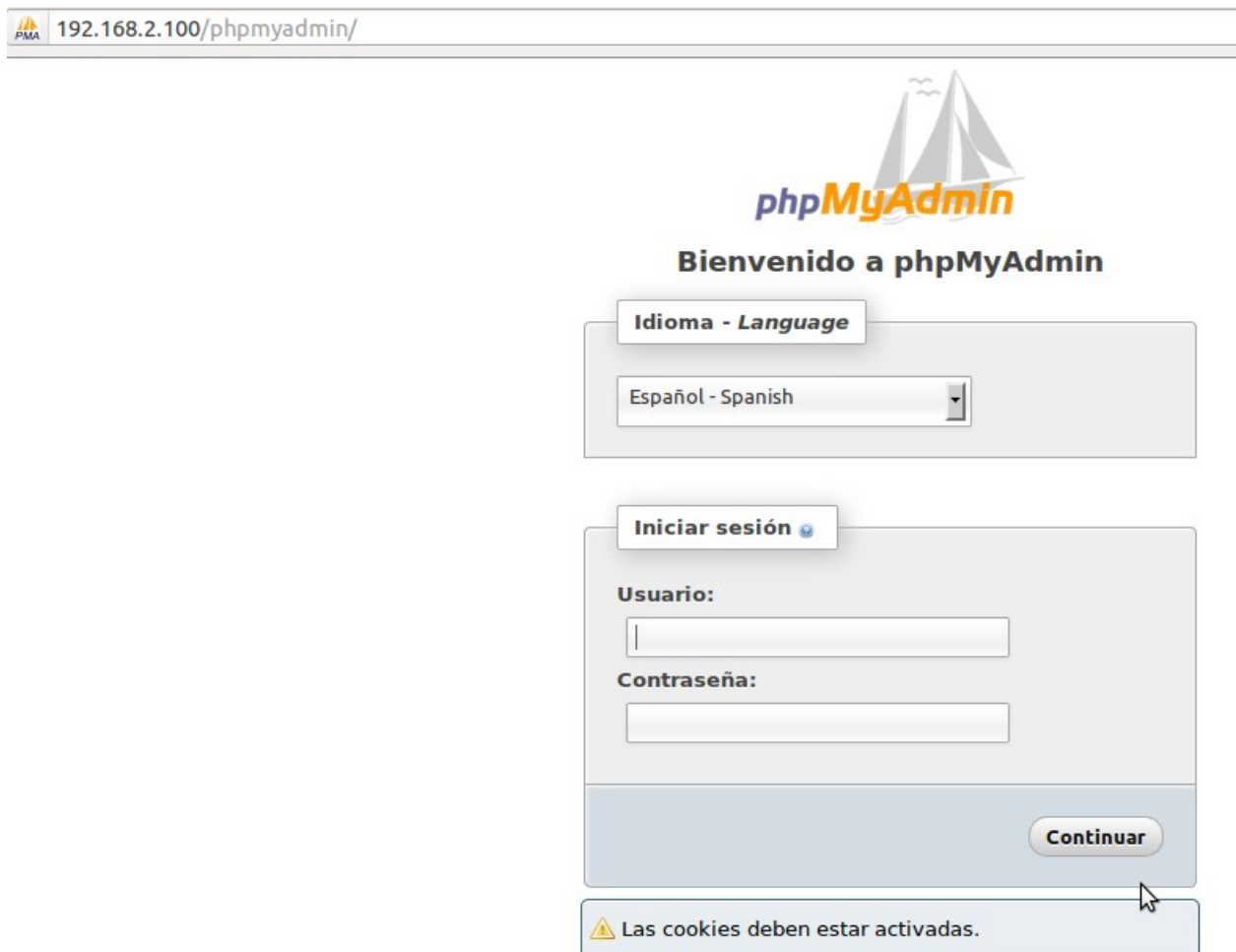
Ahora falta hacer que la carpeta del phpmyadmin apunte a la carpeta www de apache para que podamos probarlo. Para hacerlo hacemos un enlace simbólico con

```
sudo ln -s /usr/share/phpmyadmin /var/www
```

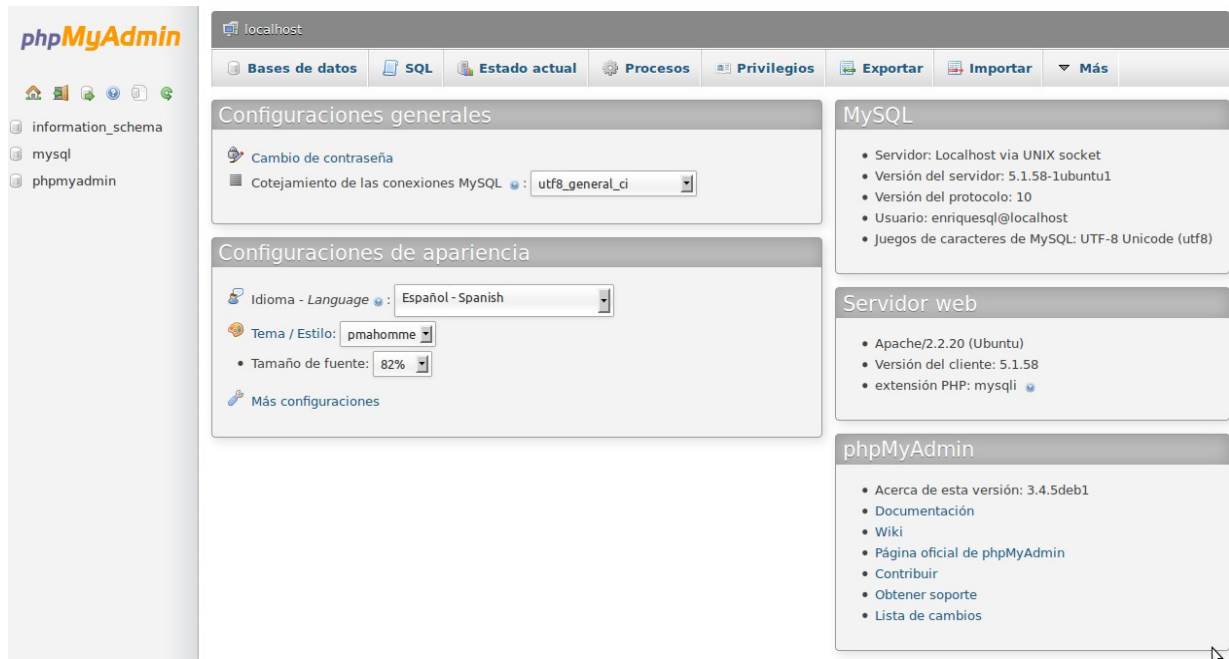
```
enrique@ServidorCep:/usr/share$ sudo ln -s /usr/share/phpmyadmin /var/www/  
[sudo] password for enrique:
```

Vamos a probarlo para ello como con los otros servicios web nos vamos a un navegador de un ordenador cliente y escribimos en la barra de dirección:

IP_servidor/phpmyadmin



Ingresamos el usuario y la contraseña que creamos anteriormente (“enriquesql” y “0000”) y accederemos al **panel de control de PHPMyAdmin**. También podemos usar el usuario root con su clave y entraríamos como administrador del sistema.



En **Privilegios** podemos ver los usuarios y sus permisos en el sistema para comprobar que el usuario que creamos tiene todo los privilegios que vamos necesitar para manejar nuestras bases de datos.



Vista global de usuarios

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z [Mostrar todo]						
Usuario	Servidor	Contraseña	Privilegios globales	Conceder	Acción	
<input type="checkbox"/> debian-sys-maint	localhost	Sí	ALL PRIVILEGES	Sí	Editar los privilegios	Exportar
<input type="checkbox"/> enriquesql	%	Sí	ALL PRIVILEGES	Sí	Editar los privilegios	Exportar
<input type="checkbox"/> phpmyadmin	localhost	Sí	USAGE	No	Editar los privilegios	Exportar
<input type="checkbox"/> root	127.0.0.1	Sí	ALL PRIVILEGES	Sí	Editar los privilegios	Exportar
<input type="checkbox"/> root	ServidorCep	Sí	ALL PRIVILEGES	Sí	Editar los privilegios	Exportar
<input type="checkbox"/> root	localhost	Sí	ALL PRIVILEGES	Sí	Editar los privilegios	Exportar

↑ [Marcar todos](#) / [Desmarcar todos](#)

CREANDO UNA COPIA DE SEGURIDAD

En este punto del curso no estaría de más que hiciéramos una copia de seguridad de nuestra máquina virtual con el sistema servidor. Para ello buscamos el archivo de la máquina virtual en la ruta donde eligiéramos guardarlo al crear la máquina. Copiamos el archivo, o la carpeta entera de la máquina a otra partición, o disco duro externo, por si cometemos algún error que no sabemos solucionar en un futuro. Esta operación tardará

unos minutos dependiendo del espacio de disco duro que le asignarais a la máquina.



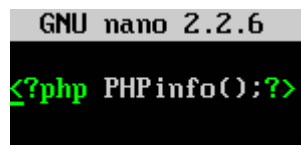
PREPARANDO LA INSTALACIÓN DE MOODLE

Antes de instalar **Moodle**, debemos ver si es compatible la versión que queremos instalar de moodle con la versión de php que estamos usando. Para eso vamos a crear un archivo con un sencillo código php **/var/www/probando2.php**.

```
sudo nano /var/www/probando2.php
```

Y vamos introducir el siguiente código

```
<? php PHPinfo ();?>
```



Guardamos el archivo. Con ese código al cargar la página en un navegador nos dará la información de la versión de php que tenemos. **En nuestro caso tenemos instalada la versión 5.3.6.**

192.168.2.100/probando2.php

**PHP Version 5.3.6-13ubuntu3.6**

System	Linux ServidorCep 3.0.0-12-generic #20-Ubuntu SMP Fri Oct 7 14:50:42 UTC 2011 i686
Build Date	Feb 11 2012 01:45:38
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/apache2
Loaded Configuration File	/etc/php5/apache2/php.ini
Scan this dir for additional .ini files	/etc/php5/apache2/conf.d
Additional .ini files parsed	/etc/php5/apache2/conf.d/gd.ini, /etc/php5/apache2/conf.d/mcrypt.ini, /etc/php5/apache2/conf.d/mysql.ini, /etc/php5/apache2/conf.d/mysqli.ini, /etc/php5/apache2/conf.d/pdo.ini, /etc/php5/apache2/conf.d/pdo_mysql.ini
PHP API	20090626
PHP Extension	20090626
Zend Extension	220090626
Zend Extension Build	API220090626,NTS
PHP Extension Build	API20090626,NTS
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
Registered PHP	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp,

Para ver la versión de MySQL basta con cargar la página de PHPMYAdmin como hicimos anteriormente.

Una vez que hemos comprobado en la página oficial de moodle (<http://download.moodle.org/>) que nuestro servidor soportará la versión de moodle que queremos instalar vamos a configurar algunos parámetros de Apache para que tenga como página por defecto además de index.html, e index.htm, también **index.php**. Para eso tenemos que editar el archivo de apache **/etc/apache2/sites-available/default**

```
sudo nano /etc/apache2/sites-available/default
```

Insertando la línea

```
DirectoryIndex index.php index.html index.htm
```

Quedando así

```

<VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DirectoryIndex index.php index.html index.htm
    DocumentRoot /var/www
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /var/www/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
    <Directory "/usr/lib/cgi-bin">
        AllowOverride None
        Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
        Order allow,deny
        Allow from all
    </Directory>

```

Para que cuando reiniciamos apache no nos aparezca más el siguiente mensaje:

```

apache2: Could not reliably determine the server's fully qualified domain name
using 127.0.1.1 for ServerName

```

Vamos a darle el nombre de **FQDN** (*Fully Qualified Domain Name*) "localhost" con el comando

```
echo "ServerName localhost" | sudo tee /etc/apache2/conf.d/fqdn
```

Y reiniciamos apache sin interrumpir las conexiones establecidas con el comando

```
sudo apache2ctl graceful
```

Comprobamos que apache se está ejecutando correctamente en nuestro sistema

```
sudo apache2ctl status
```

```

enrique@ServidorCep:~$ sudo apache2ctl graceful
enrique@ServidorCep:~$ sudo apache2ctl status
Apache Server Status for localhost

Server Version: Apache/2.2.20 (Ubuntu) PHP/5.3.6-13ubuntu3.6 with Suhosin-Patch
Server Built: Nov 7 2011 22:48:05

-----

Current Time: Thursday, 16-Feb-2012 23:10:34 CET
Restart Time: Thursday, 16-Feb-2012 22:17:30 CET
Parent Server Generation: 2
Server uptime: 53 minutes 4 seconds
Total accesses: 41 - Total Traffic: 32 kB
CPU Usage: u.02 s.1 cu0 cs0 - .00377% CPU load
.0129 requests/sec - 10 B/second - 799 B/request
1 requests currently being processed, 4 idle workers

W_____.
.
.
.
.

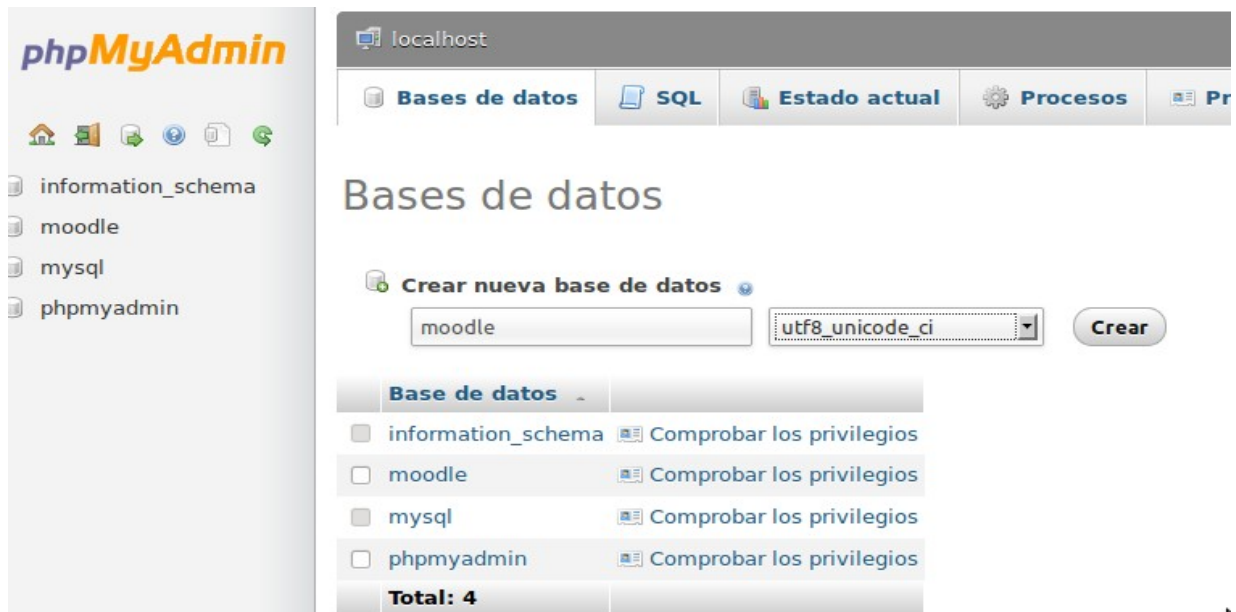
Scoreboard Key:
"_" Waiting for Connection, "S" Starting up, "R" Reading Request,
"W" Sending Reply, "K" Keepalive (read), "D" DNS Lookup,
"C" Closing connection, "L" Logging, "G" Gracefully finishing,
"I" Idle cleanup of worker, "." Open slot with no current process
enrique@ServidorCep:~$

```

CREANDO UNA BASE DE DATOS PARA MOODLE

Lo siguiente que vamos a hacer es **crear la base de datos que moodle utilizará**. Para hacerlo tenemos varias opciones con comandos, o usando PHPMyAdmin. Ya que tenemos instalado esta última herramienta vamos a usarla desde algún equipo cliente o el host anfitrión (que también es cliente claro).

Cargamos PHPMyAdmin en un navegador (ip_servidor/phpmyadmin) y a continuación nos vamos a la pestaña **Bases de Datos**. Una vez allí seleccionamos el tipo **"utf8_unicode_ci"** y el le ponemos de nombre **moodle**. Pulsamos en el botón de "Crear".



Para ver los usuarios que tienen privilegios sobre la base de datos que hemos creado, le damos a **“Comprobar privilegios”** y vemos como el usuario que creamos en esta unidad (**enriquesql**) tiene **permisos totales** en la base de datos.

DESCARGANDO MOODLE

Ya tenemos lo necesario para instalar moodle ahora es el momento de descargarlo. Tal vez la versión más extendida de moodle es la 1.9, pero nosotros vamos a instalar la última en aparecer en el momento de escribir este curso que es la 2.2.1.

Current stable builds			
Version	Information	.tgz	.zip
Moodle 2.2.1+ MOODLE_22_STABLE	This package is built every week with new fixes produced by our stable development process . It contains a number of fixes made since the 2.2.1 release and is usually a better choice for production than the actual 2.2.1 package below.		
Built Weekly 21 hours 37 mins ago	<ul style="list-style-type: none"> Recent changes log Upgrading notes Requires: PHP 5.3.2, MySQL 5.0.25 or Postgres 8.3 or MSSQL 2005 or Oracle 10.2 	Download 23.8MB 528 today	Download 23.9MB 1099 today

Para bajarla nos vamos a a la carpeta **/var/www** de nuestro servidor y usamos el comando **wget**

wget <http://downloads.sourceforge.net/project/moodle/Moodle/stable22/moodle-latest-22.tgz>

Ocupa unos 24MB así que dependiendo de la velocidad de descarga de nuestra conexión tardará más, o menos.


```

enrique@ServidorCep:/var/www$ sudo wget http://downloads.sourceforge.net/project/moodle/Moodle/stable22/moodle-latest-22.tgz
--2012-02-17 19:22:12-- http://downloads.sourceforge.net/project/moodle/Moodle/stable22/moodle-latest-22.tgz
Resolviendo downloads.sourceforge.net... 216.34.181.59
Conectando con downloads.sourceforge.net[216.34.181.59]:80... conectado.
Petición HTTP enviada, esperando respuesta... 302 Found
Ubicación: http://heanet.dl.sourceforge.net/project/moodle/Moodle/stable22/moodle-latest-22.tgz [siguiente]
--2012-02-17 19:22:13-- http://heanet.dl.sourceforge.net/project/moodle/Moodle/stable22/moodle-latest-22.tgz
Resolviendo heanet.dl.sourceforge.net... 193.1.193.66, 2001:770:18:aa40::c101:c142
Conectando con heanet.dl.sourceforge.net[193.1.193.66]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 24905356 (24M) [application/x-tar]
Grabando a: ■moodle-latest-22.tgz■

100%[=====>] 24.905.356  98,6K/s  en 3m 58s

2012-02-17 19:26:12 (102 KB/s) - ■moodle-latest-22.tgz■ guardado [24905356/24905356]

```

comprobamos que lo hemos descargado y lo **descomprimimos**.

```

enrique@ServidorCep:/var/www$ ls -l
total 76448
drwxr-xr-x  2 root root    4096 2012-02-17 19:33 html
-rw-r--r--  1 root root    177 2012-02-01 00:34 index.html
-rw-r--r--  1 root root 29416701 2012-02-17 02:41 moodle-2.1.2.zip
-rw-r--r--  1 root root 23938102 2012-02-17 02:41 moodle-2.1.4.tgz
-rw-r--r--  1 root root 24905356 2012-02-17 19:34 moodle-latest-22.tgz
lrwxrwxrwx  1 root root     21 2012-02-16 19:09 phpmyadmin -> /usr/share/phpmyadmin
-rw-r--r--  1 root root     20 2012-02-16 22:15 probando2.php
-rw-r--r--  1 root root     26 2012-02-16 22:11 probando.php

```

sudo tar xzvf moodle-latest-22.tgz

Volvemos a comprobar que al descomprimirlo nos ha creado una carpeta llamada moodle dentro de /var/www

ls -l

```

enrique@ServidorCep:/var/www$ ls -l
total 76452
drwxr-xr-x  2 root root    4096 2012-02-17 19:33 html
-rw-r--r--  1 root root    177 2012-02-01 00:34 index.html
drwxr-xr-x 41 root root    4096 2012-02-17 02:03 moodle

```

PREPARANDO LA INSTALACIÓN DE MOODLE

Además, Moodle necesita un **directorio moodledata** para almacenar archivos. Por seguridad no debe estar accesible desde la web por lo que lo crearemos en **/var** con el siguiente comando:

```
sudo mkdir -p /var/moodledata
```

Cambiaremos los permisos para que el usuario de moodle pueda acceder al mismo

```
sudo chown -R nobody:www-data /var/moodledata
```

Y los modificamos permisos del usuario sobre el mismo

```
sudo chmod -R 0770 /var/moodledata
```

```
enrique@ServidorCep:/etc$ sudo mv moodle /var/www/  
enrique@ServidorCep:/etc$ sudo mkdir -p /var/moodledata  
enrique@ServidorCep:/etc$ sudo chown -R nobody:www-data /var/moodledata  
enrique@ServidorCep:/etc$ sudo chmod -R 0770 /var/moodledata  
enrique@ServidorCep:/etc$
```

INSTALANDO Y CONFIGURANDO MOODLE

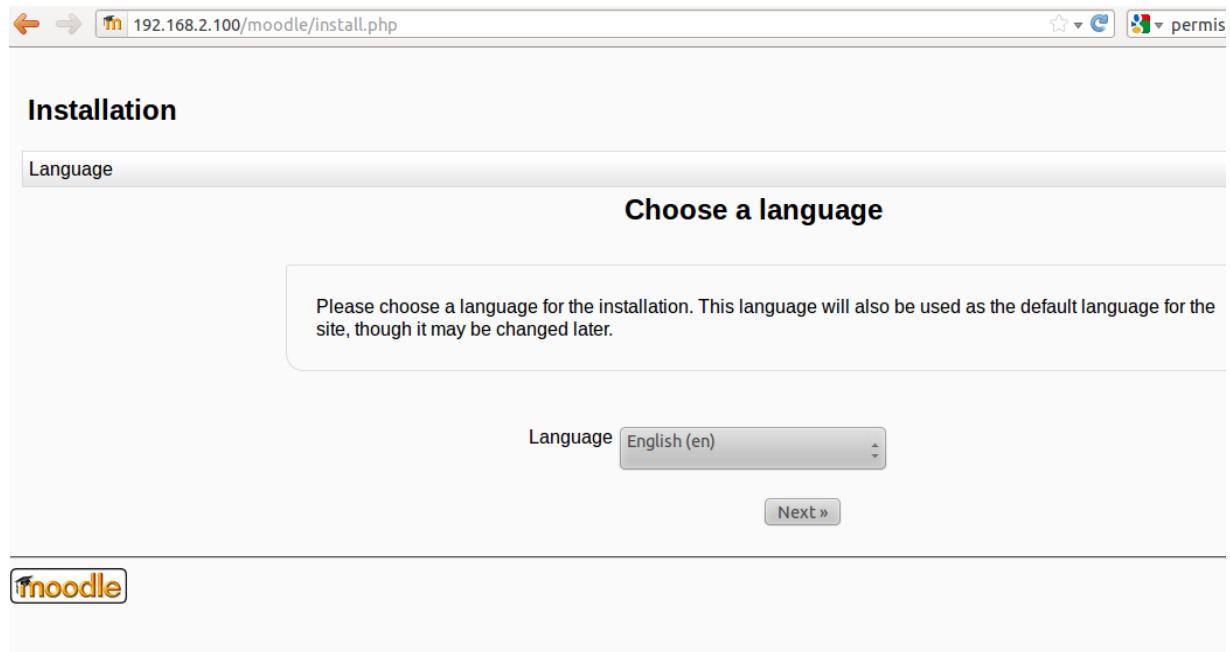
Es hora de dejar los comandos de terminal a un lado e irnos a algún equipo cliente (el equipo anfitrión por ejemplo) abriendo un navegador web y cargando la página de instalación de moodle, que si habéis seguido las mismas rutas del curso será:

http://ip_servidor/moodle

<http://192.168.2.100/moodle>

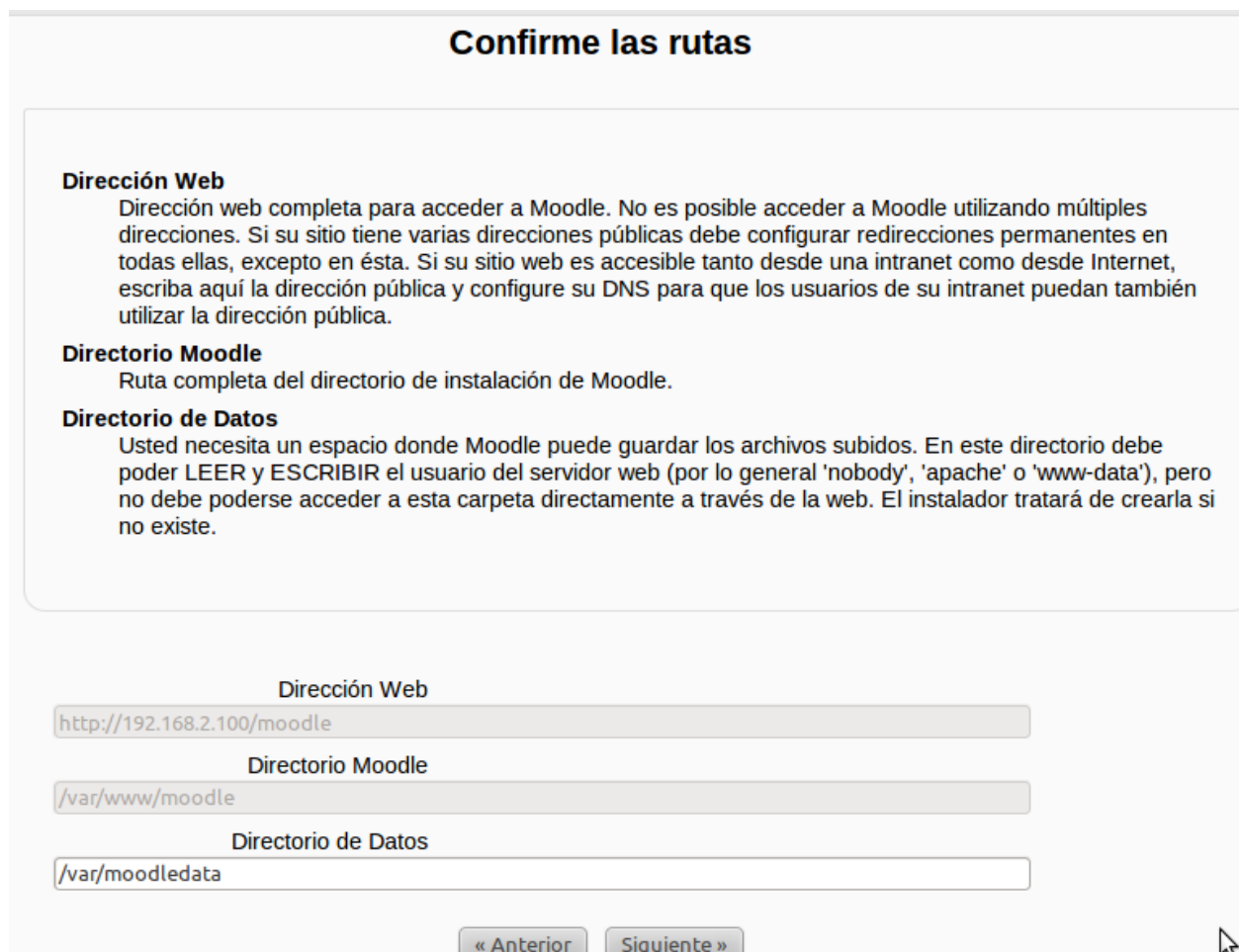
En la pantalla inicial debemos seleccionar el lenguaje, que en nuestro caso sera

Español – España (es - es)



The screenshot shows a web browser window with the address bar displaying '192.168.2.100/moodle/install.php'. The page title is 'Installation'. Below the title, there is a section labeled 'Language' with the heading 'Choose a language'. A message box states: 'Please choose a language for the installation. This language will also be used as the default language for the site, though it may be changed later.' Below this, there is a dropdown menu for 'Language' currently set to 'English (en)'. A 'Next »' button is located at the bottom right of the language selection area. The Moodle logo is visible at the bottom left of the page.

En el siguiente paso tenemos que confirmar las rutas por defecto de moodle para acceder a la web, donde se encuentran los archivos de moodle y el directorio que moodle usará para los datos.



The screenshot shows the 'Confirme las rutas' (Confirm paths) screen of the Moodle installation. It contains three sections with instructions and input fields:

- Dirección Web**: Dirección web completa para acceder a Moodle. No es posible acceder a Moodle utilizando múltiples direcciones. Si su sitio tiene varias direcciones públicas debe configurar redirecciones permanentes en todas ellas, excepto en ésta. Si su sitio web es accesible tanto desde una intranet como desde Internet, escriba aquí la dirección pública y configure su DNS para que los usuarios de su intranet puedan también utilizar la dirección pública.
Input field: `http://192.168.2.100/moodle`
- Directorio Moodle**: Ruta completa del directorio de instalación de Moodle.
Input field: `/var/www/moodle`
- Directorio de Datos**: Usted necesita un espacio donde Moodle puede guardar los archivos subidos. En este directorio debe poder LEER y ESCRIBIR el usuario del servidor web (por lo general 'nobody', 'apache' o 'www-data'), pero no debe poderse acceder a esta carpeta directamente a través de la web. El instalador tratará de crearla si no existe.
Input field: `/var/moodledata`

At the bottom, there are two buttons: « Anterior and Siguiente ».

Ahora especificamos el tipo de base de datos que nosotros usaremos que en nuestro caso es **MySQL mejorado (native/mysql)**

Base de datos

Seleccione el controlador de la base de datos

Moodle soporta varios tipos de servidores de base de datos. Por favor, póngase en contacto con el administrador del servidor si no sabe qué tipo usar.

Tipo

« Anterior Siguiente »

Configuramos los datos para acceder a nuestra base de datos donde tenemos que completar el nombre de la base de datos (**moodle**) el usuario que hemos creado (**enriquesql**) y la contraseña (**0000**) para dicho usuario.

Base de datos

Ajustes de base de datos

MySQL mejorado (native/mysql)

Ahora tiene que configurar la base de datos donde se almacenarán la mayoría de los datos de Moodle. La base de datos solo podrá crearse si el usuario de la base de datos tiene los permisos necesarios. El nombre de usuario y la contraseña ya deben existir. El prefijo de la tabla es opcional.

Servidor de la base de datos

Nombre de la base de datos

Usuario de la base de datos

Contraseña de la base de datos

Prefijo de tablas

Socket Unix ☐

« Anterior Siguiente »

En este punto moodle nos mostrará una pantalla de “**Configuración completa**”, pero nos avisará que no ha podido escribir el archivo de configuración **config.php** en la carpeta de moodle. Nos mostrará el archivo con el contenido de como debería quedar el archivo config.php que almacena la información de la configuración de nuestro moodle.

Configuración completa

Moodle ha creado su fichero de configuración.

El script de instalación no ha podido crear automáticamente un archivo config.php con las especificaciones elegidas. Por favor, copie el siguiente código en un archivo llamado config.php y coloque ese archivo en el directorio raíz de Moodle.

```
<?php // Moodle configuration file

unset($CFG);
global $CFG;
$CFG = new stdClass();

$CFG->dbtype      = 'mysqli';
$CFG->dblibrary   = 'native';
$CFG->dbhost      = 'localhost';
$CFG->dbname      = 'moodle';
$CFG->dbuser      = 'enriquesql';
$CFG->dbpass      = '0000';
$CFG->prefix      = 'mdl_';
$CFG->dboptions   = array (
    'dbpersist' => 0,
    'dbsocket'  => 0,
);

$CFG->wwwroot     = 'http://192.168.2.100/moodle';
$CFG->dataroot    = '/var/moodledata';
$CFG->admin       = 'admin';

$CFG->directorypermissions = 0777;

$CFG->passwordsaltmain = 'S4JUIwc/~`cP6Y#,fQWgi%mtcn^n';

require_once(dirname(__FILE__) . '/lib/setup.php');

// There is no php closing tag in this file,
// it is intentional because it prevents trailing whitespace problems!
```

[<< Anterior](#)
[Siguiente >>](#)

Hay varias formas de conseguir el archivo config.php con el contenido anterior pero la que me parece más sencilla es la siguiente:

CONFIGURANDO CONFIG.PHP

Nos vamos a ir a nuestro servidor a la carpeta /var/www/moodle y vamos a modificar un archivo de configuración base que viene con los archivos de moodle que descargamos y descomprimos. El archivo se llama **config-dist.php** y para editarlo usamos

sudo nano config-dist.php

Ahora vamos a ir modificándole los valores por defecto por los que nos ha reportado la pantalla de configuración completa. De esta manera tendremos que modificar el archivo

para que quede así

```
//      http://www.gnu.org/copyleft/gpl.html      //
//      //
/////////////////////////////////////////////////////////////////
unset($CFG); // Ignore this line
global $CFG; // This is necessary here for PHPUnit execution
$CFG = new stdClass();

/////////////////////////////////////////////////////////////////
// 1. DATABASE SETUP
/////////////////////////////////////////////////////////////////
// First, you need to configure the database where all Moodle data      //
// will be stored. This database must already have been created      //
// and a username/password created to access it.                      //

$CFG->dbtype      = 'mysqli';      // 'pgsql', 'mysqli', 'mssql' or 'oci'
$CFG->dblibrary    = 'native';      // 'native' only at the moment
$CFG->dbhost       = '192.168.2.100'; // eg 'localhost' or 'db.isp.com' or IP
$CFG->dbname       = 'moodle';      // database name, eg moodle
$CFG->dbuser       = 'enriquesql';  // your database username
$CFG->dbpass       = '0000';        // your database password
$CFG->prefix       = 'mdl_';        // prefix to use for all table names
$CFG->dboptions    = array(
    'dbpersist' => false,          // should persistent database connections be
                                // used? set to 'false' for the most stable
                                // setting, 'true' can improve performance

$CFG->dbtype      = 'mysqli';      // 'pgsql', 'mysqli', 'mssql' or 'oci'
$CFG->dblibrary    = 'native';      // 'native' only at the moment
$CFG->dbhost       = '192.168.2.100'; // eg 'localhost' or 'db.isp.com' or IP
$CFG->dbname       = 'moodle';      // database name, eg moodle
$CFG->dbuser       = 'enriquesql';  // your database username
$CFG->dbpass       = '0000';        // your database password
$CFG->prefix       = 'mdl_';        // prefix to use for all table names
$CFG->dboptions    = array(
    'dbpersist' => false,          // should persistent database connections be
                                // used? set to 'false' for the most stable
                                // setting, 'true' can improve performance
                                // sometimes
    'dbsocket'   => false,          // should connection via UNIX socket be used?
                                // if you set it to 'true' or custom path
                                // here set dbhost to 'localhost',
                                // (please note mysql is always using socket
                                // if dbhost is 'localhost' - if you need
                                // local port connection use '127.0.0.1')
    'dbport'     => '',            // the TCP port number to use when connecting
                                // to the server. keep empty string for the
                                // default port

);
```

```

// that your students would use. Do not include a trailing slash
//
// If you need both intranet and Internet access please read
// http://docs.moodle.org/en/masquerading

$CFG->wwwroot    = 'http://192.168.2.100/moodle';

//=====
// 4. DATA FILES LOCATION
//=====
// Now you need a place where Moodle can save uploaded files. This
// directory should be readable AND WRITEABLE by the web server user
// (usually 'nobody' or 'apache'), but it should not be accessible
// directly via the web.
//
// - On hosting systems you might need to make sure that your "group" has
//   no permissions at all, but that "others" have full permissions.
//
// - On Windows systems you might specify something like 'c:\moodledata'

$CFG->dataroot    = '/var/moodledata';

//=====
// The following parameter sets the permissions of new directories
// created by Moodle within the data directory. The format is in
// octal format (as used by the Unix utility chmod, for example).
// The default is usually OK, but you may want to change it to 0750
// if you are concerned about world-access to the files (you will need
// to make sure the web server process (eg Apache) can access the files.
// NOTE: the prefixed 0 is important, and don't use quotes.

$CFG->directorypermissions = 02777;

//=====
// 6. DIRECTORY LOCATION (most people can just ignore this setting)
//=====
// A very few webhosts use /admin as a special URL for you to access a
// control panel or something. Unfortunately this conflicts with the
// standard location for the Moodle admin pages. You can fix this by
// renaming the admin directory in your installation, and putting that
// new name here. eg "moodleadmin". This will fix admin links in Moodle.

$CFG->admin = 'admin';

//=====

```

No hará falta que modifiquemos ningún valor más. Observad que el parámetro **CFG->passwordsaltmain**, no lo hemos modificado como venía en el archivo que nos genera la instalación porque no vamos a usar ese añadido de seguridad a las contraseñas.

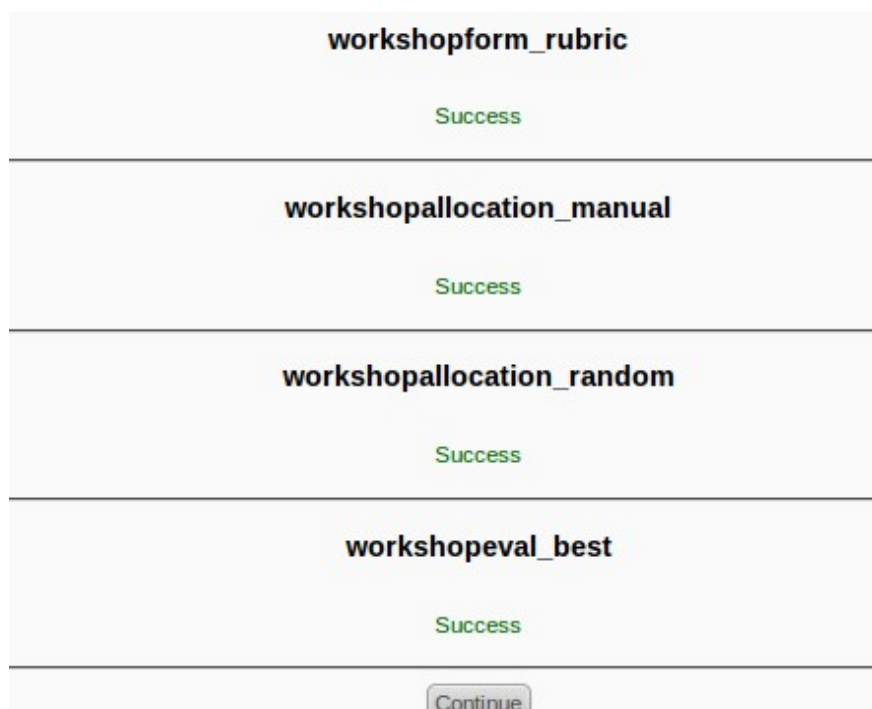
Una vez terminemos la edición **guardamos con el nombre config.php y salimos**.

Volvemos a un navegador en un equipo cliente y cargamos de nuevo la dirección

http://ip_servidor/moodle

<http://192.168.2.100/moodle>

Seguirá el proceso de instalación de los módulos de moodle hasta que nos permita pulsar el botón **“continue”**



Completamos algunos datos personales referentes al administrador y otras opciones de configuración.

Installation

On this page you should configure your main administrator account which will have complete access to the system. Please make sure you give it a secure username and password as well as a valid email address. You can change these later on.

General

Username*

Choose an authentication method **Manual accounts**

New password* ☐ **Unmask**

Force password change ☐

First name*

Surname*

Email address* **Required**

Email display

Email format

Email digest type

Forum auto-subscribe

When editing text

AJAX and Javascript

Screen reader

City/town*

Select a country*

Timezone

Preferred language

Por último configuramos los parámetros básicos de la página web de moodle, como el nombre de la página de inicio o una breve descripción de la misma.

USANDO LOS DNS

Hasta ahora hemos accedido a la página web alojada en nuestro servidor desde los clientes a través de la dirección IP del mismo. Es posible que queramos acceder a ella usando un **nombre de dominio más fácil de recordar**. Para esto tenemos dos opciones, o usamos un servidor de nombres de dominio donde esté establecida la asociación de la dirección ip al nombre que queremos ponerle a nuestra página, o podemos resolver esta asociación localmente introduciéndola en el archivo **/etc/hosts**. Para hacerlo editamos el archivo de manera que haya una entrada así:

ip_servidor nombre_dominio

192.168.2.100 www.servidorcep.es

De esta forma cuando pongamos en un navegador de un equipo cliente al que le hemos introducido esa asociación en su archivo hosts, www.servidorcep.es este cargará la página alojada en nuestro servidor como si hubiéramos introducido la dirección IP directamente.

Este artículo esta licenciado bajo Creative Commons Attribution-NonCommercial-ShareAlike 2.5 License.

Servidores Linux Enrique Brotons