



IES Luis Vélez de Guevara
Écija (Sevilla)

DESARROLLO WEB EN ENTORNO SERVIDOR

2º DAW

**Profesor: Alfonso Jiménez
Vílchez.**

Índice

Práctica 2: Instalación de la plataforma LAMP.....	3
Instalación.....	3
Creando un host virtual en un dominio de pruebas: dwes.lan.....	8
Creando un host virtual en el puerto 8888.....	10

Práctica 2: Instalación de la plataforma LAMP

Instalación

Para instalar la plataforma **LAMP** es posible instalar los componentes uno a uno, pero es más cómodo utilizar el comando **tasksel** desde la consola. Este comando viene con algunas tareas predefinidas, que nos permiten instalar con un solo comando grupos de aplicaciones. Entre las tareas que incluye **tasksel** se encuentra **lamp-server**, que incorpora los componentes de una arquitectura **LAMP** antes mencionados.

Para instalar **LAMP** ejecuta desde una consola:

sudo apt install tasksel

sudo tasksel install lamp-server

```
usuario@DAW2-PC00:~$ sudo apt install tasksel
[sudo] contraseña para usuario:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es necesario.
libllvm7
Utilice «sudo apt autoremove» para eliminarlo.
Se instalarán los siguientes paquetes adicionales:
tasksel-data
Se instalarán los siguientes paquetes NUEVOS:
tasksel tasksel-data
0 actualizados, 2 nuevos se instalarán, 0 para eliminar y 556 no actualizados.
Se necesita descargar 34,2 kB de archivos.
Se utilizarán 292 kB de espacio de disco adicional después de esta operación.
¿Desea continuar? [S/n] S
Des:1 http://es.archive.ubuntu.com/ubuntu bionic/main amd64 tasksel-data all 3.34ubuntu11 [5.476 B]
Des:2 http://es.archive.ubuntu.com/ubuntu bionic/main amd64 tasksel all 3.34ubuntu11 [28,7 kB]
Descargados 34,2 kB en 0s (128 kB/s)
Preconfigurando paquetes ...
Seleccionando el paquete tasksel-data previamente no seleccionado.
(Leyendo la base de datos ... 176243 ficheros o directorios instalados actualmente.)
Preparando para desempaquetar .../tasksel-data_3.34ubuntu11_all.deb ...
Desempaquetando tasksel-data (3.34ubuntu11) ...
Seleccionando el paquete tasksel previamente no seleccionado.
Preparando para desempaquetar .../tasksel_3.34ubuntu11_all.deb ...
Desempaquetando tasksel (3.34ubuntu11) ...
Configurando tasksel (3.34ubuntu11) ...
Configurando tasksel-data (3.34ubuntu11) ...
Procesando disparadores para man-db (2.8.3-2ubuntu0.1) ...
usuario@DAW2-PC00:~$ sudo tasksel install lamp-server
```

Para reforzar la seguridad de la instalación de mysql, ejecuta la siguiente línea:

sudo mysql_secure_installation

```
usuario@DAN2-PC00:~$ sudo mysql_secure_installation

Securing the MySQL server deployment.

Connecting to MySQL using a blank password.

VALIDATE PASSWORD PLUGIN 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 plugin?

Press y|Y for Yes, any other key for No: p
Please set the password for root here.

New password:

Re-enter new password:
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : p

... skipping.

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

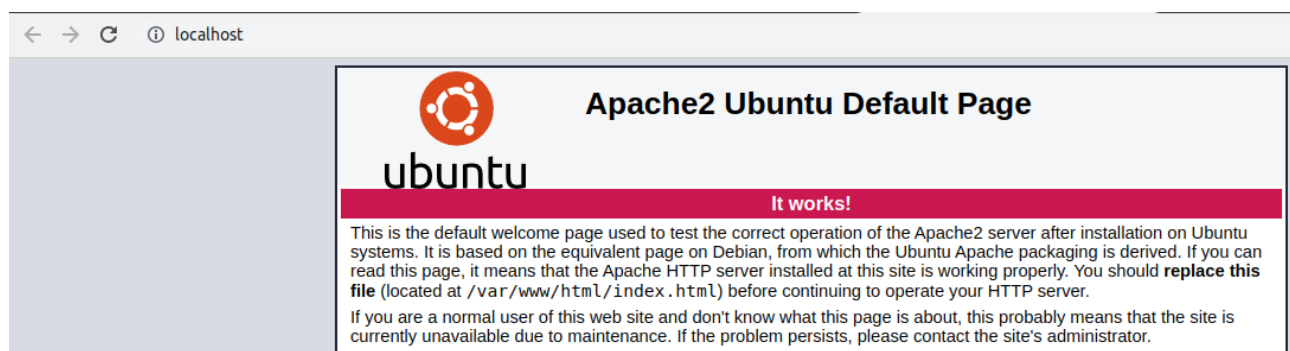
Disallow root login remotely? (Press y|Y for Yes, any other key for No) : p

... skipping.
By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : p

... skipping.
Reloading the privilege tables will ensure that all changes
```

Accede a <http://localhost> para comprobar que el servidor apache funciona.



Con la configuración predeterminada, la carpeta raíz de dónde el servidor web Apache extrae los documentos que va a publicar es **/var/www/html**.

Para probar el correcto funcionamiento de la plataforma, crea en ese directorio un fichero **“prueba.php”** con el siguiente contenido (tendrás que hacerlo como root):

```
usuario@DAW2-PC00:/$ cd var/www/html/  
usuario@DAW2-PC00:/var/www/html$ sudo nano prueba.php  
[sudo] contraseña para usuario:  
usuario@DAW2-PC00:/var/www/html$ cat prueba.php  
<?php  
phpinfo();  
?>
```

phpinfo() es una función del lenguaje PHP que genera por sí misma una página web completa con información sobre la instalación de PHP. Si todo va bien, al abrir esa página con un navegador web (desde la propia máquina de Ubuntu la URL será <http://localhost/prueba.php>) verás una página con información sobre la versión de php instalada.

Localiza en esta página la siguiente información:

•short_open_tag

short_open_tag	Off	Off
----------------	-----	-----

•max_execution_time

max_execution_time	30	30
--------------------	----	----

•error_reporting

error_reporting	22527	22527
-----------------	-------	-------

•file_uploads

file_uploads	On	On
--------------	----	----

•upload_max_filesize

upload_max_filesize	2M	2M
---------------------	----	----

•Ruta del archivo php.ini que se está utilizando.

Loaded Configuration File	/etc/php/7.2/apache2/php.ini
---------------------------	------------------------------

Y por último instalamos **phpMyAdmin**:

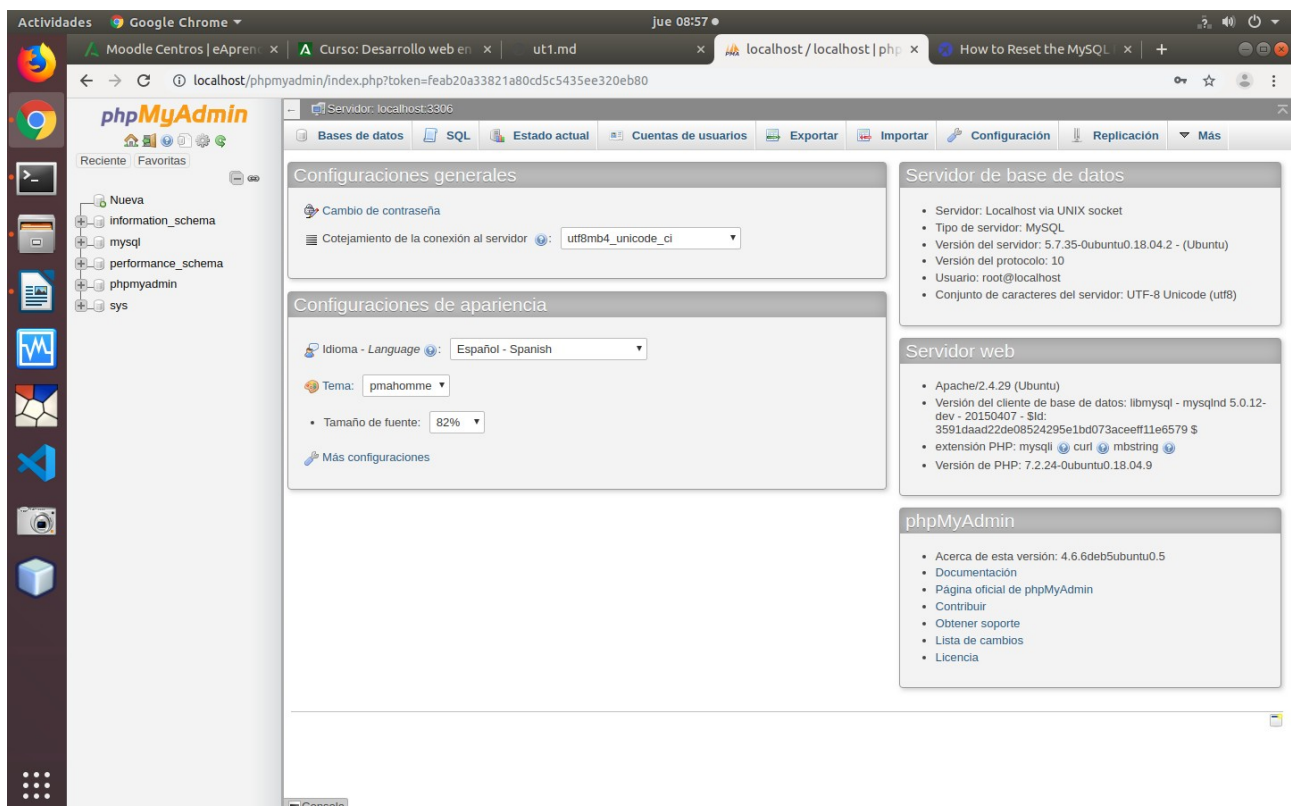
sudo apt-get install phpmyadmin

En el proceso de instalación nos preguntará si queremos reconfigurar apache o lighttpd. Escogemos apache pulsando espacio para marcar la casilla correspondiente.

```
usuario@DAW2-PC08:~$ sudo apt-get 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 javascript-common libcurl4 libjs-jquery libjs-sphinxdoc libjs-underscore libzip4 php php-bz2 php-curl php-gd
php-mbstring php-pear php-php-gettext php-phpseclib php-tcpdf php-xml php-zip php7.2 php7.2-bz2 php7.2-curl php7.2-gd php7.2-mbstring php7.2-xml
php7.2-zip
Paquetes sugeridos:
php-libsodium php-mcrypt php-gmp php-imagick
Se instalarán los siguientes paquetes NUEVOS:
dbconfig-common dbconfig-mysql javascript-common libcurl4 libjs-jquery libjs-sphinxdoc libjs-underscore libzip4 php php-bz2 php-curl php-gd
php-mbstring php-pear php-php-gettext php-phpseclib php-tcpdf php-xml php-zip php7.2 php7.2-bz2 php7.2-curl php7.2-gd php7.2-mbstring php7.2-xml
php7.2-zip phpmyadmin
```

Accede a <http://localhost/phpmyadmin> y comprueba que funciona, iniciando sesión con el usuario root y la contraseña que has configurado. Si falla prueba a ejecutar las siguientes líneas (sustituyendo la contraseña por la deseada):

sudo mysql -u root



Una vez tenemos el servidor montado, la manera más inmediata de empezar a trabajar es copiando nuestros programas a la carpeta /var/www/html, creando incluso carpetas si es necesario (como hemos hecho con el archivo prueba.php). De esa manera tenemos todo accesible en <http://localhost/> o <http://127.0.0.1/> o <http://xxx.xxx.xxx.xxx> (mi ip) desde

2º DAW , Juan Miguel Costa Lucena, 22/09/2021.

cualquier otro equipo de la red. Sin embargo, hay una forma más elegante y organizada de trabajar: creando hosts virtuales (vhosts) en los que montar nuestros sitios web.

Creamos una carpeta para los scripts de DWES en /var/www/dwes

sudo mkdir -p /var/www/dwes

```
usuario@DAW2-PC00:~$ sudo mkdir -p /var/www/dwes  
[sudo] contraseña para usuario:
```

```
usuario@DAW2-PC00:/var/www$ tree  
.  
├── dwes  
└── html  
    ├── index.html  
    └── prueba.php
```

La hemos creado como root, pero para trabajar comodamente con ella le asignamos nuestro usuario como dueño:

sudo chown -R \$USER:\$USER /var/www/dwes

```
usuario@DAW2-PC00:~$ sudo chown -R $USER:$USER /var/www/dwes
```

Le damos permiso de lectura y ejecución al resto de usuarios para que Apache pueda leer los archivos:

sudo chmod -R 755 /var/www/dwes

```
usuario@DAW2-PC00:~$ sudo chmod -R 755 /var/www/dwes
```

Creemos en /var/www/dwes un archivo index.html de bienvenida:

```
<html>  
  <head><title>Prueba UT1</title></head>  
  <body>  
    <h1>Hola mundo, soy TU_NOMBRE</h1>  
  </body>  
</html>
```

```
usuario@DAW2-PC00:~$ nano /var/www/dwes/index.html  
usuario@DAW2-PC00:~$ cat /var/www/dwes/index.html  
<html>  
  <head><title>Prueba UT1</title></head>  
  <body>  
    <h1>Hola mundo, soy Juanmi</h1>  
  </body>  
</html>
```

Para crear un host virtual tenemos que crear los archivos de configuración que le indican a apache qué directorio contiene las páginas y en qué puerto, dominio... tiene que escuchar.

Para ello, en **/etc/apache2/sites-available** podemos encontrar los archivos de configuración de los vhosts (sitios web) disponibles.

Como base para crear el nuestro, copiamos el archivo de configuración por defecto que incluye apache:

sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/dwes.conf

```
usuario@DAW2-PC00:~$ sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/dwes.conf
```

Creando un host virtual en un dominio de pruebas: dwes.lan

Si queremos crear un host virtual en <http://dwes.lan> editamos el archivo `/etc/apache2/sites-available/dwes.conf` para que quede de la siguiente manera (cambiando el correo electrónico por el nuestro):

<VirtualHost *:80>

ServerAdmin webmaster@localhost

DocumentRoot /var/www/dwes

ServerName dwes.lan

ServerAlias www.dwes.lan

ErrorLog \${APACHE_LOG_DIR}/error.log

CustomLog \${APACHE_LOG_DIR}/access.log combined

</VirtualHost>

```
usuario@DAW2-PC00:/etc/apache2$ sudo nano sites-available/dwes.conf
```

```
usuario@DAW2-PC00:/etc/apache2$ cat sites-available/dwes.conf
```

```
<VirtualHost *:80>
```

```
    ServerAdmin jmcostalucena2000@gmail.com
```

```
    DocumentRoot /var/www/dwes
```

```
    ServerName dwes.lan
```

```
    ServerAlias www.dwes.lan
```

```
    ErrorLog ${APACHE_LOG_DIR}/error.log
```

```
    CustomLog ${APACHE_LOG_DIR}/access.log combined
```

```
</VirtualHost>
```


Una vez hecho esto, activamos el host virtual y reiniciamos apache:

sudo a2ensite dwes.conf

sudo systemctl reload apache2

```
usuario@DAW2-PC00:~$ sudo a2ensite dwes.conf
Enabling site dwes.
To activate the new configuration, you need to run:
    systemctl reload apache2
usuario@DAW2-PC00:~$ sudo systemctl reload apache2
```

Por último, nos queda "engañar" al sistema para que piense que dwes.lan es nuestro servidor, para ello editamos el archivo /etc/hosts:

sudo nano /etc/hosts

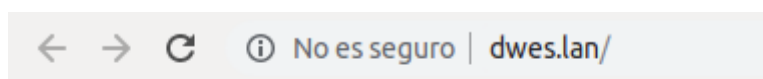
Y añadimos la siguiente línea:

127.0.0.1 dwes.lan

```
usuario@DAW2-PC00:~$ sudo nano /etc/hosts
usuario@DAW2-PC00:~$ cat /etc/hosts
127.0.0.1      dwes.lan
127.0.0.1      localhost
127.0.1.1      DAW2-PC00

# 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
```

De esta manera, cuando accedamos a <http://dwes.lan> , podremos ver nuestra página web.



Hola mundo, soy Juanmi

Si queremos desactivar el host virtual, ejecutamos:

sudo a2dissite dwes.conf

sudo systemctl reload apache2

```
usuario@DAW2-PC00:~$ sudo a2dissite dwes.conf
Site dwes disabled.
To activate the new configuration, you need to run:
    systemctl reload apache2
usuario@DAW2-PC00:~$ sudo systemctl reload apache2
```

Al ejecutar **a2dissite**, el sistema borra el enlace para nuestro archivo de configuración de la carpeta **/etc/apache2/sites-enabled**. De esta manera el archivo de configuración no se pierde aunque el sitio esté desactivado.

Creando un host virtual en el puerto 8888

Si queremos crear un host virtual en el puerto 8888 editamos el archivo **/etc/apache2/sites-available/dwes.conf** para que quede de la siguiente manera (cambiando el correo electrónico por el nuestro):

La linea listen obliga a Apache a estar atento al puerto indicado

Listen 8888

<VirtualHost *:8888>

ServerAdmin webmaster@localhost

DocumentRoot /var/www/dwes

ErrorLog \${APACHE_LOG_DIR}/error.log

CustomLog \${APACHE_LOG_DIR}/access.log combined

</VirtualHost>

```
usuario@DAW2-PC00:~$ cat /etc/apache2/sites-available/dwes.conf
Listen 8888
<VirtualHost *:8888>

    ServerAdmin jmcostalucena2000@gmail.com
    DocumentRoot /var/www/dwes
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Una vez hecho esto, activamos el host virtual y reiniciamos apache:

sudo a2ensite dwes.conf

sudo systemctl reload apache2

Al ejecutar a2ensite, el sistema crea un enlace para nuestro archivo de configuración en la carpeta **/etc/apache2/sites-enabled**. Así es como Apache sabe, de los sitios web que tenemos configurados en el equipo, cuáles son los que están activos.

De esta manera, cuando accedamos a <http://localhost:8888> , podremos ver nuestra página web.



Si queremos desactivar el host virtual, ejecutamos:

sudo a2dissite dwes.conf

sudo systemctl reload apache2

Al ejecutar a2dissite, el sistema borra el enlace para nuestro archivo de configuración de la carpeta **/etc/apache2/sites-enabled**. De esta manera el archivo de configuración no se pierde aunque el sitio esté desactivado.