

INSTITUTO TECNOLOGICO SUPERIOR DE MISANTLA

SERVIDOR WEB APACHE

Servicios de Red

Autor: José Luis Barrientos Terán

**domingo, 24 de
octubre de 2010**

Docente: Ing. Raul Mario Santos Alvarez

INDICE

<i>Instalando apache</i> _____	2
<i>Publicar contenido desde el directorio personal de usuarios</i> _____	3
<i>Autenticación en directorios por usuario y contraseña</i> _____	4
Evitar el listado de directorios de un servidor web _____	6
<i>Instalación de Joomla</i> _____	7

Instalando apache

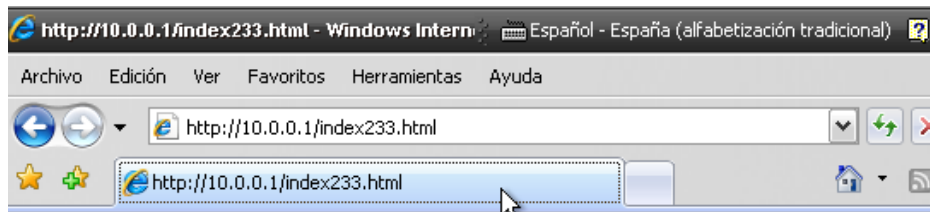
El servidor web Apache, uno de los mas usados, nos permite tener directorios individuales para los usuarios del sistema en los que pueden tener su página web o sus archivos y acceder a ellos desde un navegador de internet sin necesidad de estarlos metiendo en el directorio raiz del servidor.

Esto significa que cada usuario del sistema dentro de su directorio home puede crear una carpeta llamada `public_html` y crear sus páginas web guardando sus archivos en la carpeta mencionada, y en un momento veremos como habilitar esta funcionalidad de apache.

Para instalar apache con soporte para php y mysql en Ubuntu 10.04 se necesita la siguiente instrucción:

```
root@bubuntuServer:~# apt-get install apache2 php5 mysql-client mysql-server
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
```

Podemos comprobar en el navegador de algún cliente tecleando la dirección ip del servidor. Por ejemplo <http://10.0.0.1>



It works!

This is the default web page for this server.

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

Publicar contenido desde el directorio personal de usuarios

Para activar este modulo se debe hacer un link de los archivos “userdir.conf” y “userdir.load” desde el directorio mods-available al directorio mods-enabled , estos directorios se encuentran en “/etc/apache/”.

Los dos comandos para activar este modulo son:

```
root@bubuntuServer:~#  
root@bubuntuServer:~# ln -s /etc/apache2/mods-available/userdir.conf /etc/apache  
2/mods-enabled/userdir.conf
```

```
root@bubuntuServer:~#  
root@bubuntuServer:~# ln -s /etc/apache2/mods-available/userdir.load /etc/apache  
2/mods-enabled/userdir.load _
```

Despues de esto tenemos que reinicar el servidor web de esta forma:

```
root@bubuntuServer:~#  
root@bubuntuServer:~# /etc/init.d/apache2 restart  
* Restarting web server apache2  
... waiting [ OK ]  
root@bubuntuServer:~# _
```

Por ultimo creamos la carpeta llamada “public_html” desde nuestro directorio personal en este caso logeados solamente como administrador y hacemos una pagina en html dentro de este directorio para visualizarla.

Autenticación en directorios por usuario y contraseña

El método de autenticación que utilizaremos será el HTTP Digest, el servicio apache esta conformado por diversos módulos que por razones practicas y para hacer a este servicio mas liviano no viene con todos activados, una de las ventajas de usar este método es que es fácil de implementar y tanto el login como la contraseña van encriptados, pero una desventaja es que no todos los navegadores Web lo soporta.

Para comenzar vamos a editar el archivo 000-default y agregaremos unas líneas con el nombre del directorio a proteger:

```
root@bubuntuServer:~#  
root@bubuntuServer:~# nano /etc/apache2/sites-enabled/000-default
```

```
# alert, emerg.  
LogLevel warn  
  
CustomLog /var/log/apache2/access.log combined  
  
Alias /doc/ "/usr/share/doc/"  
<Directory "/usr/share/doc/">  
  Options Indexes MultiViews FollowSymLinks  
  AllowOverride None  
  Order deny,allow  
  Deny from all  
  Allow from 127.0.0.0/255.0.0.0 ::1/128  
</Directory>  
  
<Directory "/var/www/cosas/" >  
  AuthName "zonaprivate"  
  AuthType Digest  
  AuthUserFile /etc/apache2/passwords.conf  
  Require valid-user  
</Directory>  
</VirtualHost>
```

A continuación la explicación de este código:

AuthName: indica el nombre del dominio de autenticación.

AuthType: indica que el método a usar es Digest.

AuthDigestDomain: dominio protegido con autenticación digest.

AuthUserFile: indica donde se encuentra el archivo de contraseñas que ahora llamamos .htdigest.

Require: usuarios que tienen acceso a los recursos especificados.

valid-user: cualquier usuario incluido en el archivo de contraseñas .htpasswd

Luego de editar el archivo debemos guardar los cambios **ctrl + o** y salimos **ctrl + x**, después procedemos a crear el archivo de contraseñas de la siguiente manera:

```
root@bubuntuServer:~# htdigest -c /etc/apache2/passwords.conf zonaprivate luis_
```

Donde:

1- El parámetro victor.org debe ser el mismo que agregamos en **AuthName** porque se cuándo se crea un usuario se hace con un dominio concreto.

2- Si el parámetro Require indica '**valid-user**', se consideran usuarios válidos sólo los que pertenecen al dominio de autenticación dado en AuthName y las contraseñas sólo pueden utilizarse en este dominio.

3- En el parámetro **nom_usuario** colocamos el nombre de usuario que queramos.

4- La carpeta **passwd** la creamos previamente en el directorio **/etc/apache2/**

5- igual la podemos llamar como queramos lo único que hay que tener en cuenta es, a la hora de generar la clave colocarla con el nombre correcto, lo mismo pasaría con el archivo de claves .htdigest

Por ultimo solo queda reiniciar el servicio y comprobar que exista dicha autenticación.



Evitar el listado de directorios de un servidor web

Para evitar el listado solo tendremos que agregar una línea al archivo de configuración del sitio por default esto lo hacemos con los siguientes comandos:

```
root@bubuntuServer:~#  
root@bubuntuServer:~#  
root@bubuntuServer:~# nano /etc/apache2/sites-enabled/000-default
```

Se busca este trozo de código y se agrega la siguiente línea:

```
<Directory /var/www/>  
Options -Indexes FollowSymLinks MultiViews  
AllowOverride none  
Order allow,deny  
allow from all  
</Directory>
```

Por ultimo volvemos a reiniciar el servidor y probamos borrando el index de la pagina principal. Nos tendrá que salir un mensaje de error.



Instalación de Joomla

Los pasos para instalar Joomla sobre Ubuntu Server son los siguientes:

1. Nos aseguramos de encontrarnos en nuestro *home*:
`cd`
2. Creamos un directorio para almacenar la estructura de Joomla:
`mkdir joomla`
3. Nos cambiamos al nuevo directorio:
`cd joomla`
4. Descargamos la última versión en español de Joomla (en el momento de escribir este tutorial es la 1.5.14):
`wget`
`http://joomlancode.org/gf/download/frsrelease/12979/56159/Joomla_1.5.21-Spanish-pack_completo.zip`
5. Descomprimos el archivo descargado:
`unzip Joomla_1.5.21-Spanish-pack_completo.zip`
6. Borramos el archivo comprimido porque ya no lo volveremos a necesitar:
`rm -f Joomla_1.5.21-Spanish-pack_completo.zip`
7. Nos cambiamos al directorio superior para poder mover correctamente, en el siguiente paso, el directorio *joomla* que era en el que nos encontrábamos:
`cd ..`
8. Movemos el directorio que contiene la estructura de Joomla a */var/www*:
`sudo mv joomla /var/www`
9. Para hacer que nuestro servidor sea algo más seguro vamos a cambiar el propietario y el grupo de todos los archivos y directorios (de forma recursiva) de */var/www/joomla*:
`sudo chown -R root:root /var/www/joomla`
10. Nos cambiamos al directorio */var/www/joomla*:
`cd /var/www/joomla`
11. Creamos la variable **LUGARES** (se escribe tal como esta en la terminal):
`LUGARES='`
`administrator/backups`
`administrator/components`
`administrator/modules`
`administrator/templates`
`cache`
`components`
`images`
`images/banners`
`images/stories`
`language`
`plugins`
`media`
`modules`


```
templates
```

12. Para todos los directorios de la variable *LUGARES* cambiamos el usuario y grupo propietarios a *www-data* (el usuario de Apache):

```
for i in $LUGARES; do
sudo chown -R www-data:www-data $i
done
```

13. Cambiamos los permisos de los archivos para que todos los usuarios puedan leerlos pero sólo el propietario tenga permisos de escritura:

```
sudo find . -type f -exec chmod 644 {} \;
```

14. Cambiamos los permisos de los directorios para que todos los usuarios puedan leerlos y *ejecutarlos* pero sólo el propietario pueda escribir en ellos:

```
sudo find . -type d -exec chmod 755 {} \;
```

15. Creamos una base de datos para Joomla de nombre *bdjoomla* (pero puedes ponerle el nombre que quieras):

```
mysqladmin -u root -p create bdjoomla
```

16. Accedemos a MySQL con el usuario *root* y la contraseña que hubiésemos elegido cuando lo instalamos:

```
mysql -u root -p
```

y creamos un usuario de nombre *joomlaadmin* y contraseña *nimdaalmooj* (también puedes cambiar estos nombres):

```
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX, ALTER, CREATE
TEMPORARY TABLES, LOCK TABLES ON bdjoomla.* TO 'joomlaadmin'@'localhost'
IDENTIFIED BY 'nimdaalmooj';
```

17. Activamos los permisos:

```
FLUSH PRIVILEGES;
```

18. Salimos de MySQL:

```
quit
```

19. Eliminamos el historial de MySQL porque la contraseña se lee a simple vista:

```
rm -f ~/.mysql_history
```

20. Por fin, ya estamos preparados para empezar la instalación. Así que, como en principio suponemos que estamos sobre un Ubuntu Server, nos vamos a otro equipo de la red que tenga interfaz gráfica y abrimos un navegador. En la dirección escribimos <http://10.0.0.1/joomla> (cambia 10.0.0.1 por la dirección IP de tu servidor).

21. Entonces veremos un asistente para ayudarnos a instalar Joomla en el que lo primero que debemos hacer es elegir el idioma de la instalación. Y después hacemos clic en **Siguiente**.



Seleccionamos el idioma

22. En el siguiente paso consiste en una comprobación previa a la instalación para que sepamos lo bien o mal que tenemos preparado el sistema. En principio, no hace falta que todos los parámetros estén en verde pero nosotros sí vamos a hacer todo lo necesario para que así sea. Por eso, nos fijamos que tenemos dos advertencias en rojo que vamos a solucionar.



Comprobamos si los valores son los correctos

23. Para solucionar el primero de los problemas creamos el archivo `/var/www/joomla/configuration.php`, que es donde Joomla guardará los datos de configuración de nuestro sitio web:

sudo touch /var/www/joomla/configuration.php

24. Le cambiamos el propietario y el grupo para poder escribir en él:

sudo chown www-data:www-data /var/www/joomla/configuration.php

25. Y nos aseguramos que tenemos permiso de escritura asignándole de nuevo los permisos:

sudo chmod 644 /var/www/joomla/configuration.php

26. Pasamos a solucionar el segundo de los problemas detectados. Por eso editamos el archivo `/etc/php5/apache2/php.ini` con el comando:

```
sudo nano /etc/php5/apache2/php.ini
```

y cambiamos el valor de la directiva `display_errors` de `On` a `Off`:

```
display_errors = Off
```

27. Continuamos con la instalación haciendo clic sobre el botón **Volver a comprobar** (dependiendo del tiempo que haya pasado nos aparecerá un mensaje de error y deberemos comenzar de nuevo). Lo primero que hacemos es verificar que ahora todos los parámetros son correctos y están en verde:



Comprobamos que los parámetros son correctos

28. En el siguiente paso leemos la licencia y, una vez hayamos finalizado, hacemos clic sobre **Siguiente**.



Leemos la licencia

29. A continuación, debemos escribir los datos de configuración de MySQL.

Utilizaremos el usuario `joomlaadmin` creado específicamente para administrar la base de datos `bdjoomla`. Y pulsamos **Siguiente**:



Configuramos la base de datos

30. Para no hacer más largo este tutorial no configuraremos el servidor FTP. Así que hacemos clic en **Siguiente**:



No activamos el FTP

31. En la configuración principal deberemos escribir tanto el nombre de nuestro sitio web como el correo electrónico del administrador y la contraseña para el usuario *admin*. El usuario *admin* lo crea Joomla por defecto y será con el que empezemos a administrar nuestro sitio. Además podemos instalar datos de ejemplo pulsando sobre el botón correspondiente:



Escribimos los datos de configuración principales

32. ¡Listo! Ya hemos terminado de instalar Joomla. Se nos indica que borremos el directorio *installation* porque si no lo hacemos no podremos usar Joomla y se nos recuerda que el nombre de usuario para entrar en la interfaz de administración es *admin*.



Hemos terminado la instalación de Joomla!

33. De nuevo en el servidor, una vez que hemos terminado la instalación debemos eliminar el directorio `/var/www/joomla/installation` como se nos indicó en el paso anterior:

```
sudo rm -R /var/www/joomla/installation/
```

34. Y no debemos olvidarnos de modificar los permisos del archivo `configuration.php` para que sea de solo lectura para todo el mundo:

```
sudo chmod 444 /var/www/joomla/configuration.php
```

35. Ahora desde cualquier equipo de nuestra red abrimos un navegador y escribimos la dirección `http://10.0.0.1/joomla` para acceder a nuestro nuevo portal.

Accedemos a nuestro sitio Joomla!

36. Si lo que queremos es administrar nuestro Joomla, deberemos utilizar la siguiente dirección: `http://192.168.1.5/joomla/administrator`



Control de acceso a la zona de administración de Joomla!

37. Y después de introducir nuestro nombre de usuario (*admin*) y la contraseña que escribimos, tendremos acceso a la nuestra interfaz de administración:



Accedemos a la interfaz de administración de Joomla

Sin embargo, nos queda una última cosa por hacer, cambiar la ruta del directorio temporal. Para esto hacemos clic en el icono **Configuración**.

38. En la pantalla de **Configuración global**, hacemos clic sobre **Servidor** y en la **Ruta al directorio temporal** escribimos **/tmp**. Después guardamos los cambios y listo.



Cambiamos la ruta del directorio temporal