

Instalación de XAMPP para linux

Las siguientes indicaciones se refieren a un sistema Ubuntu, aunque son válidas, o fácilmente adaptables, para cualquier sistema linux.

Lo primero que necesitamos es el paquete XAMPP, lo podemos descargar desde la página de apachefriends.org [1] donde podemos encontrar la última versión estable, así como información sobre dicho paquete. Una vez en la página indicada, nos desplazamos hacia la zona de descarga y en este caso nos vamos a bajar el fichero para instalar en Linux para sistemas de 64 bits.



Una vez terminada la descarga podemos realizar la comprobación de la integridad del archivo mediante la comprobación del md5. Situando el cursor del ratón sobre "md5" de la columna checksum correspondiente a la versión descargada, nos aparece el hash del archivo. Podéis utilizar la utilidad GtKHash que os permitirá realizar dicha comprobación (la podéis encontrar en el centro de software de Ubuntu, o buscar ayuda en internet).

Una vez hemos calculado y comprobado el hash md5 podemos proceder a la instalación de XAMPP.

Abrimos un terminal.

Nos desplazamos al directorio donde se encuentra el archivo que hemos descargado.

En este caso el archivo descargado se llama `xampp-linux-x64-5.6.12-0-installer.run` y es un ejecutable. XAMPP se instala en el directorio `/opt` por lo que vamos a necesitar permisos de administrador. Ejecutamos el siguiente comando `sudo ./xampp-linux-x64-5.6.12-0-installer.run` (modificar el nombre del archivo por el nombre del archivo descargado). Con esto se nos abrirá el instalador de XAMPP.



Hacemos click en el botón "Next", vamos leyendo las instrucciones que nos irán apareciendo y haciendo click en los botones "Next" correspondientes hasta que lleguemos al final de la instalación.



Cuando la instalación haya terminado se nos indicará en la ventana, así como la posibilidad de iniciar el panel del control de XAMPP, con el que podremos Iniciar / Detener los distintos servidores que trae XAMPP (en esta versión: servidor apache, servidor mysql, servidor FTP).

Configuración de Linux y Apache para conseguir comportamiento web en local

Con lo visto en la sección anterior podemos utilizar nuestra máquina como un servidor web con tecnología PHP, y acceso a BBDD (Bases de Datos) pero de momento debemos acceder a las páginas creadas desde el navegador como `http://localhost/carpeta-de-proyecto/ruta-pagina-web.html` (p.e.)

Lo que pretendemos lograr es poder acceder a nuestro proyecto tal y como lo haríamos con un servidor en internet, `http://www.mi-dominio.local`

Fijate que el TLD del dominio (.com .es .en ...) lo he puesto como .local, la razón de esto no es mero capricho, sino porque lo que vamos a hacer a continuación es decirle a nuestra máquina y a Apache es que este dominio se encuentra en nuestra máquina, y si pusieramos un TLD como p.e. .com para un dominio en producción existente en internet no podríamos acceder a las páginas situadas en internet, ya que nuestro ordenador siempre las buscaría en local.

Considero que la elección de .local como costumbre es adecuada ya que la relación de su situación es inmediata, sobre todo si utilizamos el dominio .local para pruebas de un dominio en producción en internet, antes de aplicar los cambios.

Empezamos con la configuración de Linux:

Abrimos un terminal y ejecutamos `sudo gedit` con lo que abriremos un editor con permisos de administrador. Ahora podremos editar archivos del sistema y modificarlos. Desde gedit abrimos el archivo `/etc/hosts` y añadimos los siguientes cambios por cada uno de los dominios que queramos utilizar de forma local



A la izquierda una IP 127.0.0.1 que representa a la máquina en la que estamos localhost, le estamos diciendo que el dominio que indicamos en la derecha debe buscarlo en la propia maquina sin salir a internet.

```
127.0.0.1 www.dominio.tld
```

Añadimos también la línea sin (www.), para que así resuelva más rápido si utilizamos el nombre del dominio sin www

```
127.0.0.1 dominio.tld
```

y ahora archivo->guardar
ya podemos cerrar gedit.

Ahora debemos configurar apache

Decidimos en que carpeta vamos a guardar nuestros proyectos, por ejemplo www en nuestra carpeta personal, una en la que tengamos acceso de lectura y escritura. (p.e.:/home/usuario/www)

Primero modificamos el archivo de configuración para poder utilizar hosts virtuales

Ejecutamos `sudo gedit` y en archivo->abrir localizamos el archivo:

```
/opt/lampp/etc/httpd.conf
```

Buscamos las líneas donde se indican

```
<Directory "/opt/lampp/htdocs">
```

...

...

</Directory>



Copiamos las líneas (Las líneas que se ven en la imagen) entre

```
<Directory "/opt/lampp/htdocs">
```

y el primer

```
</Directory>
```

ambas incluidas

las duplicamos

y en las líneas nuevas que acabamos de duplicar, sustituimos `/opt/lamm/htdocs` por `/home/usuario/www` quedando un bloque con `/opt/lamm/htdocs`

```
<Directory "/opt/lampp/htdocs">
```

...

...

```
</Directory>
```

y otro bloque con `/home/usuario/www`

```
<Directory "/home/usuario/www">
```

...

...

```
</Directory>
```

Ahora buscamos las líneas

```
# Virtual hosts
```

```
# Include etc/extra/httpd-vhosts.conf
```

y descomentamos la segunda (le quitamos el caracter #), quedando como sigue

```
# Virtual hosts
```

```
Include etc/extra/httpd-vhosts.conf
```

Guardamos y podemos cerrar gedit

Por último debemos configurar en que directorio se guardarán los archivos para cada dominio local (dentro de /home/usuario/www).

Ejecutamos `sudo gedit` y en archivo->abrir localizamos el archivo:

```
/opt/lampp/etc/extra/httpd-vhosts.conf
```

comentamos los ejemplos que incluye el archivo, (colocamos el caracter # al inicio de cada linea)

Cada dominio virtual comienza con

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

y termina con

```
</VirtualHost>
```

por cada dominio local que necesitemos debemos incluir un bloque, con almenos la siguiente información, cambiando la información adecuada para vuestras necesidades. Creamos uno adicional para localhost.

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

```
    DocumentRoot "/opt/lampp/htdocs"  
    ServerName localhost
```

```
</VirtualHost>
```

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

```
    DocumentRoot "/home/javier/www/lopeztorrijos"  
    ServerName lopeztorrijos.local  
    ServerAlias www.lopeztorrijos.local
```

```
</VirtualHost>
```

guardamos el archivo y podemos cerrar `gedit`. Cuando lo necesitemos ahora ya podemos iniciar apache desde el panel de control de XAMPP.

Iniciar y detener los servidores de XAMPP

Para iniciar y detener los servidores de XAMPP (Apache, MySQL, ProFTPD) tenemos varias opciones:

- Abrimos un terminal y ejecutamos `sudo /opt/lampp/lampp start` para iniciar y `sudo /opt/lampp/lampp stop` para detener. En el terminal se nos muestran los distintos cambios de estado de los servidores. La opción indicada en el apartado FAQs de la página web que se nos muestra escribiendo localhost en la barra de direcciones de nuestro navegador, si tenemos los servidores de XAMPP iniciados.
- Ejecutamos `sudo /opt/lampp/manager-linux-x64.run` (el nombre del ejecutable puede cambiar

dependiendo de la versión instalada) desde un terminal. Se nos abrirá el panel de control para así poder iniciar/detener los distintos servidores.



- Creamos un lanzador cuyo comando sea `gksu /opt/lampp/manager-linux-x64.run` (el nombre del ejecutable puede cambiar dependiendo de la versión instalada), para abrir el panel de control. Utilizamos `gksu` (lo podeis instalar desde el centro de software), ya que así se nos mostrará una ventana solicitando la contraseña de administrador necesaria.



Francisco Javier López Torrijos
Analista Sistemas Informáticos de Gestión
Diseño y Desarrollo Web

URL de origen (modified on 09/28/2016 - 00:19): <http://www.lopeztorrijos.com/node/62>

Enlaces

[1] <http://www.apachefriends.org/en/xampp-windows.html>