

## Instalar XAMPP como servidor local

Vamos a instalar XAMPP como servidor local. Lo que vamos a realizar en este tutorial es instalar en nuestra máquina XAMPP un paquete en el cual se incluye:

- Servidor Apache -- servidor páginas web HTML, PHP, ASP
- Servidor MySQL -- servidor de bases de datos
- Servidor FileZilla -- servidor FTP
- Servidor Mercury -- servidor de correo
- Servidor Tomcat -- servidor páginas web Java

Con este paquete podemos hacer, en nuestro caso, que nuestro equipo actúe tal y como lo haría un servidor web para de esta manera poder realizar todas las instalaciones y pruebas que necesitemos, para la realización de páginas web utilizando distintas tecnologías

## Instalación de XAMPP para Win

Lo primero que necesitamos es el paquete XAMPP, lo podemos descargar desde la página de [apachefriends.org](http://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 Win



a la izquierda resaltado con un cuadrado rojo el fichero que nos descargamos, y a la derecha el hash md5 del fichero, que utilizaremos para comprobar la integridad del fichero una vez descargado. Podéis descargar WinMD5 desde [aquí](#) [2], simplemente una vez instalado buscamos el instalador de XAMPP descargado, y calculamos su hash md5, comparamos el hash con el que se nos indica en la página de descarga de XAMPP, debiendo coincidir.

Una vez comprobado el hash doble-click sobre el instalador y seguimos las instrucciones que nos vayan apareciendo en las ventanas de diálogo. (windows puede pedirnos permiso para la instalación, aceptamos si es el caso)



seleccionamos la ubicación donde queremos instalar nuestro XAMPP.



marcamos las casillas de verificación según nuestras preferencias y click en install para proceder con la instalación, una vez concluida la instalación nos aparece una ventana de cmd con algunas opciones, como iniciar el panel de control, cambiar la ubicación, .....

Ya tenemos instalado el paquete, haciendo doble-click en el icono de XAMPP, en el escritorio, si le hemos dicho que queríamos que lo crease y se iniciará el panel-de-control de XAMPP desde el cual podemos iniciar:

- Apache que será lo que habitualmente necesitaremos, tanto para páginas HTML, PHP
- MySQL si necesitamos acceso a Bases de Datos (habitual para CMS,CRM como Drupal, Joomla, WordPress, sugarCRM ....)

## Configuración de Win 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>

Fíjate 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 pusiéramos 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 Win:

Iniciamos el bloc de notas de Win, si es necesario vamos a inicio->buscamos:bloc de notas . click-derecho sobre el programa->Ejecutar como administrador



En Win7 debemos editar el archivo como administrador para que nos permita modificarlo ya que es un fichero del sistema, sino cuando le digamos guardar simplemente no lo hará y no servirá de nada.

Desde el block de notas, vamos a archivo->abrir->c:->windows->sistem32->drivers->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

y ahora archivo->guardar  
ya podemos cerrar el block de notas.

~~Para que esta última modificación surta efecto debemos reiniciar Win pero todavía no habremos terminado.~~

## Ahora debemos configurar apache

?

Abrimos el block de notas, en esta ocasión no es necesario ejecutarlo como administrador, y en archivo->abrir localizamos el archivo:

C:->ruta-instalación-xampp->apache->conf->extra->http-vhosts.conf

debemos añadir un bloque como el que aparece en la siguiente imagen, poniendo los valores adecuados, los añadiremos al final del archivo para asegurarnos de no ponerlo dentro de otro bloque por error.  
Dentro del propio fichero hay ejemplos con mas opciones



Una vez incluido este bloque, deberemos buscar una línea en el mismo fichero como la siguiente, como mínimo,

```
##NameVirtualHost *:80
```

y descomentarla, osea borrar las # quedando asi:

```
NameVirtualHost *:80
```

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

## ¿Donde situar los archivos de nuestro proyecto?

Ahora veremos donde situar los archivos de nuestros proyectos para que el servidor Apache los sirva Desde el explorador de archivos de win localizamos

C:->carpeta-instalación-xampp->htdocs

y creamos una carpeta por cada proyecto (dominio) que necesitemos, la cual contendrá todos los archivos y carpetas necesarias en dicho proyecto.



podemos visitar nuestra web en local, desde cualquier navegador web poniendo en la barra de direcciones, suponiendo un **tld .local** p.e.:

<http://www.tu-dominio.local>

# Comprobar y configurar seguridad en los servidores locales

Ahora nos falta comprobar y configurar la seguridad de nuestros servidores locales

Hacemos doble-click sobre el acceso directo que se ha creado en el escritorio



Se nos abre el panel de control e iniciamos pulsando sobre el botón **start** tanto el **servidor apache** como el servidor **MySQL**



Si todo ha ido bien los botones pasan a **stop** y nos aparecerá **Running** a la izquierda de cada uno. Para acceder a la página de **localhost** o servidor local, tenemos dos opciones:

Se abre una página en la que podemos seleccionar el idioma, seleccionamos español en nuestro caso

- Hacer click sobre el botón **admin** correspondiente al servidor **Apache** lo que nos abrirá el navegador por defecto de nuestro sistema y accederá a **localhost**
- Abrir nuestro navegador y en la barra de direcciones poner **localhost** + intro.



Se nos abrirá una página de bienvenida en la cual se nos da algo de información. En esta se nos dice que vayamos a la entrada de **estado** para que veamos que componentes están en funcionamiento. Los servidores de FTP y correo no estén en funcionamiento, de momento no los vamos a utilizar. Luego vamos a la entrada de **checkeo de seguridad**



Se nos da información de algunos problemas de seguridad iniciales, Básicamente el Administrador de MySQL (**root**) **no tiene password** y el servidor **Apache** tampoco, si leemos la información ofrecida, veremos un enlace para poder subsanar estos fallos de seguridad, hacemos click en él.

Vamos a una página en la que se nos pide una contraseña para el administrador de **MySQL (root)**, apuntala y no la pierdas pues te hará falta para acceder al servidor MySQL así como para iniciarlo.



También podemos asignar el usuario y contraseña para el administrador de **Apache**, igual que con MySQL apunta los datos de acceso y no los pierdas pues te los pedirá la próxima vez que inicies **Apache**



Debemos hacer primero uno y luego el otro ya que solo se actualiza uno cada vez.

Si ahora volvemos a comprobar el **chequeo de seguridad** de nuestros servidores veremos que los que nos interesan se han solucionado



También podemos comprobar el **estado** y veremos que los que de momento nos interesan están bien excepto **MySQL** esto es porque no hemos accedido como root de MySQL, enseguida lo haremos



Para comprobar que todo ha ido bien cerramos el navegador, y lo volvemos a abrir, accedemos a **localhost** desde la barra de direcciones y nos aparecerá un popup solicitando el usuario y contraseña para **Apache**, Recuerda el que cambiamos antes y debiste apuntar



Bien pues ahora solamente nos falta acceder al servidor de **MySQL** buscamos a la izquierda la entrada que pone **phpMyAdmin** hacemos click sobre el enlace y se nos abrirá una página en la que se nos pide el usuario y contraseña para acceder a MySQL, recuerda también debiste apuntarte estos datos de acceso. los ponemos y los confirmamos



Ahora ya podemos administrar nuestras Bases de Datos MySQL, pero esto deberá ir en otro tutorial.

Recuerda no hemos configurado ni arrancado el servidor FTP ni el de correo, por eso podemos ver algunos fallos sobre estos servidores.

Tampoco hemos solucionado el hecho de que PHP no funcione en **safe mode** o modo seguro, ya que tal y como nos indica en la explicación del fallo de seguridad si vamos a utilizar el servidor para desarrollo algunas funcionalidades dejarían de funcionar

Para tus pruebas puedes utilizar el usuario **root** de MySQL para acceder a la BBDD pero recuerda que no es seguro para ambientes de producción, además cuando contrates un servidor para tus sitios web no te darán acceso para el usuario **root** y con permisos totales sino con permisos de acceso restringidos.

archivo:

[Icono de PasajeNotApache1.png](#)[3]

## 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. Podeis utilizar la utilidad GtKHash que os permitirá realizar dicha comprobación (la podreis 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 harimos 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 máquina 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 qué 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 carácter #), 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 05/21/2013 - 22:32):** <http://www.lopeztorrijos.com/node/20>

**Enlaces**

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

[2] <http://www.winmd5.com/>

[3] [http://www.lopeztorrijos.com/sites/default/files/xamppPassRootApache1\\_0.png](http://www.lopeztorrijos.com/sites/default/files/xamppPassRootApache1_0.png)