

# Guía del Administrador del Sistema

## Tabla de contenidos

- [Guía del Administrador del Sistema](#)
  - [Configuración PHP](#)
    - [Manipulación de php.ini](#)
    - [Ajustes necesarios](#)
    - [Ajustes recomendados](#)
  - [Configuración MySQL](#)
    - [Un usuario de MySQL por aplicación web](#)
  - [Establecimiento de autenticación básica \(.htaccess\)](#)
  - [Hacer su instalación de PrestaShop más segura](#)
  - [Afinamiento y rendimiento](#)
    - [archivo config.inc.php](#)
    - [archivo defines.inc.php](#)
    - [archivo smarty.inc.php](#)
  - [Mejoramiento del rendimiento de PrestaShop](#)
  - [Otras recomendaciones](#)
    - [Modo seguro](#)
    - [Actualizaciones](#)
  - [Misceláneo](#)
    - [Estructura de archivos PrestaShop](#)
    - [Traslado de PrestaShop](#)

## Guía del Administrador del Sistema

Esta guía le ayudará a configurar un servidor web para que sea mejor y más seguro.

Una vez realizada esta configuración, usted estará listo para instalar PrestaShop, utilizando nuestra [Introducción](#).

### Configuración PHP

#### Manipulación de php.ini

Muchos de los consejos en esta guía requieren editar el archivo `php.ini` que se encuentra en la carpeta de instalación de PHP de su servidor (no en la carpeta de PrestaShop).

No todos los host le permitirán editar o incluso acceder a este archivo, así que contacte con ellos si no puede acceder a él.

Por ejemplo, es probable que no obtenga acceso a `php.ini` en un hosting compartido. Si su host no proporciona la configuración requerida por defecto y usted no puede manipular `php.ini`, entonces debe cambiarse a un hosting dedicado o uno más permisivo.

Aun así, editar `php.ini` sigue siendo una acción técnica y avanzada. Si su negocio funciona bien en la actualidad, no hay necesidad de modificar ese archivo y mucho menos cambiar de host.

La edición de la configuración PHP requiere que cambie algunos valores en el archivo `php.ini` la mayoría de las veces de "On" a "Off" o viceversa. El archivo contiene una gran cantidad de documentación por cada línea: asegúrese de leer la perteneciente a sus cambios, con el fin de entenderlos mejor. Tenga cuidado con lo que edita, ya que esto tiene un impacto directo en la forma como PHP se ejecuta y por lo tanto, en la estabilidad y la seguridad de sus servidores.

#### Ajustes necesarios

Para que PrestaShop 1.4.x funcione correctamente, su instalación de PHP debe contener los siguientes ajustes y bibliotecas:

- MySQL.
- Biblioteca GD.
- Extensión Dom.
- allow\_url\_fopen.

La extensión MySQL permite acceder a sus datos. PrestaShop simplemente no puede funcionar sin él.

La biblioteca GD permite a PHP manipular imágenes de forma dinámica. PrestaShop la utiliza para cambiar el tamaño y reelaborar archivos de imágenes que son subidos (marcas de agua, limpieza, etc.). Sin imágenes, una tienda virtual pierde gran interés, ¡asegúrese que GD se encuentre habilitado!

La extensión Dom permite analizar documentos XML. PrestaShop la utiliza para diversas funciones, como el Localizador de Tiendas. También es utilizada por algunos módulos, así como la biblioteca `pear_xml_parser`.

La directiva `allow_url_fopen` permite a los módulos acceder a archivos remotos, los cuales son una parte esencial del proceso de pago, entre otras cosas. Por lo tanto, es imprescindible que se encuentre configurado en **ON**.

En definitiva, es imprescindible contar con las siguientes directivas establecidas en los valores indicados:

```
extension = php_mysql.dll
extension = php_gd2.dll
allow_url_fopen = On
```

## Ajustes recomendados

Su instalación de PHP debe contener los siguientes ajustes y bibliotecas, para mejor experiencia:

- Soporte GZIP.
- Biblioteca Mcrypt.
- `register_globals` deshabilitado.
- `magic_quotes` deshabilitado.
- `allow_url_include` deshabilitado.

Contar con el soporte GZip permite al servidor web reunir las páginas web, las imágenes y secuencias de comandos antes de enviarlos al navegador. Esto hace que navegar por la tienda sea más rápido y por lo tanto sea una experiencia más agradable.

Mcrypt ofrece a PHP con una capa de seguridad endurecida, permite el uso de más funciones hash y algoritmo de criptografía.

La directiva `register_globals` cuando se encuentra habilitada, define todas las variables de entorno (GET, POST, COOKIE, SERVIDOR...) como variables globales. **No es seguro usar variables no definidas**, ya que un usuario pudiera establecer un valor a esta variable utilizando el método GET, por ejemplo. Por lo tanto, es fundamental configurar `{register_globals en OFF`.

La directiva `magic_quotes` automáticamente escapa (o "agrega antislash") a todas las secuencias de caracteres especiales (' , " , \ , NULL) para todas las variables de entorno (GET, POST, COOKIE, SERVIDOR...). Esta opción se debe configurar en **OFF** ya que agregará slash a cada variable, incluso aquellas que no necesitan ser agregadas. En muchos casos, algunas aplicaciones Web pasan por alto esta opción, por lo que algunas variables podrían ser agregadas dos veces dando como resultado datos dañados.


La directiva `allow_url_include` se utiliza para permitir la inclusión de cualquier archivo a través de declaraciones `require` y `include`, aunque no provengan de su servidor Web. Esta opción debe ser configurada en **OFF**, ya que si una aplicación en el servidor web sufre de "inclusión de vulnerabilidad", los usuarios podrán incluir cualquier archivo desde cualquier servidor y estos se ejecutarán en su propio servidor.

En resumen, **se recomienda** tener las siguientes directivas establecidas en los valores indicados:

```
register_globals = Off
magic_quotes_gpc = Off
allow_url_include = Off
```

## Configuración MySQL

MySQL a menudo tiene una cuenta de administrador por defecto ("root", "admin"...), que ofrece acceso a todo el contenido de las bases de datos, sin importar quién la gestione. El administrador tiene todos los derechos y puede realizar todas las acciones posibles. Por lo tanto, necesita "salvaguardar" sus bases de datos, para evitar que sus aplicaciones web sucumban a las [inyecciones de SQL](#) (puede ocurrir cuando un usuario tiene éxito en la obtención de la contraseña de administrador).

 Si acaba de instalar MySQL, agregue una contraseña para la cuenta root, la cual no tiene contraseña por defecto.

### Un usuario de MySQL por aplicación web

Cada vez que instale una nueva aplicación web en su servidor, debe crear un nuevo usuario MySQL, con los derechos necesarios para manejar los datos de dicha aplicación. NO utilice el mismo nombre de usuario que usa para manejar las bases de datos en todas las aplicaciones web instaladas.

Si tiene acceso a una cuenta principal de MySQL, que puede crear otros usuarios, lo podría realizar utilizando la línea de comandos:

```
mysql -u USERNAME -p PASSWORD
```

También podría utilizar la siguiente consulta SQL:

```
mysql> USE mysql;
mysql> CREATE USER 'username'@'servername' IDENTIFIED BY 'new_password';
```

Tenga en cuenta que su host podría ofrecerle acceso a una herramienta en línea para realizar tareas de administración de MySQL con más facilidad, como cPanel. Utilícela, ya que es probable que no contará con acceso a la línea de comandos en ese caso.

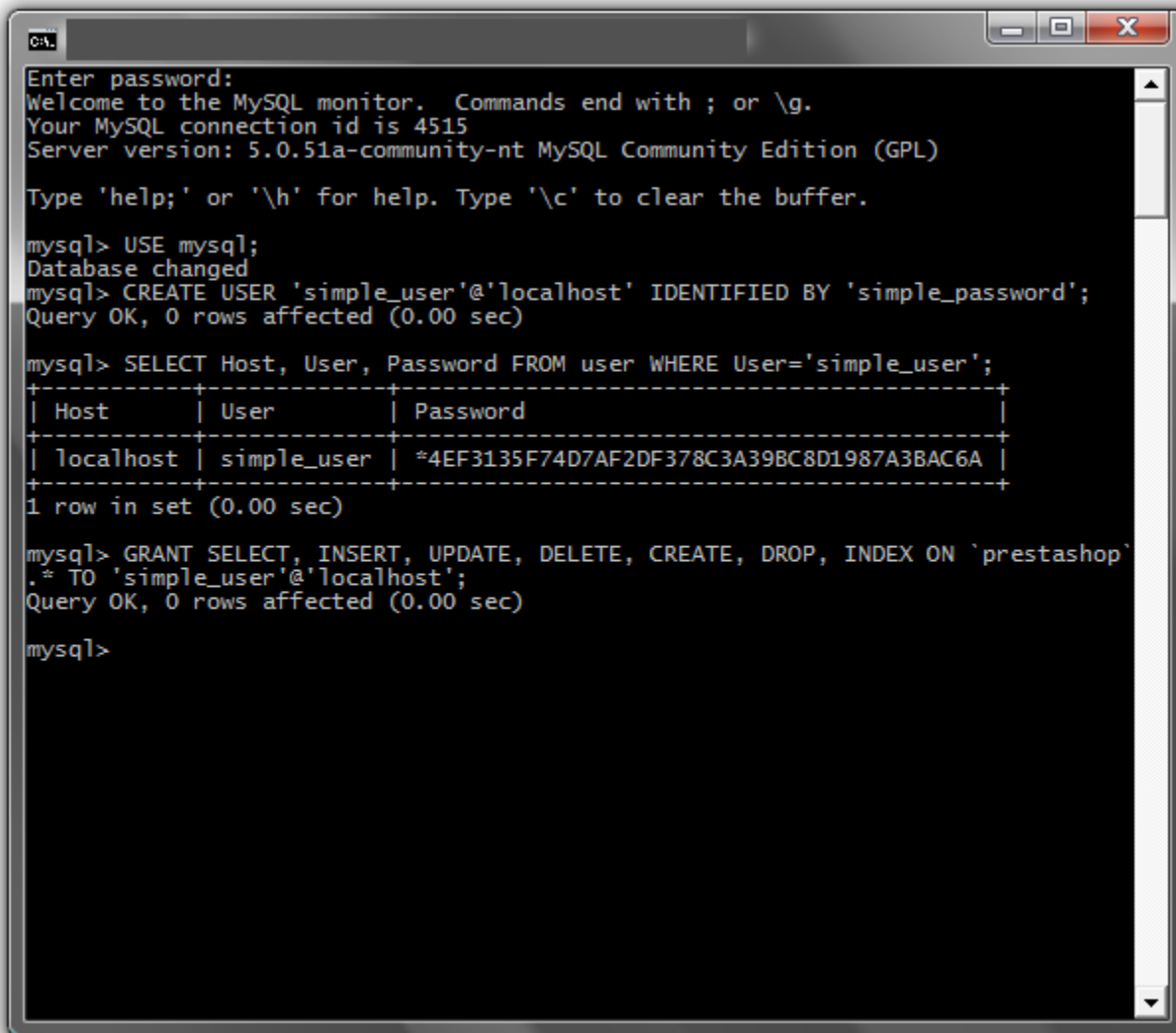
Ahora contamos con un nombre de usuario con los derechos suficientes para conectarse a la base de datos local.

Tenemos que permitir a este usuario utilizar la base de datos "prestashop" y configurar sus derechos al mismo tiempo. Aquí le mostramos una plantilla para la consulta SQL para realizarlo:

```
mysql> GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, ALTER
> ON 'prestashop'.* TO 'new_user'@'localhost';
mysql> FLUSH PRIVILEGES;
```

Ahora tenemos un usuario sólo para nuestra base de datos "prestashop". Recuerde realizar esto para cada nueva aplicación web que agregue a su servidor.

Ahora puede instalar PrestaShop en forma segura.



```
C:\>
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 4515
Server version: 5.0.51a-community-nt MySQL Community Edition (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> USE mysql;
Database changed
mysql> CREATE USER 'simple_user'@'localhost' IDENTIFIED BY 'simple_password';
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT Host, User, Password FROM user WHERE User='simple_user';
+-----+-----+-----+
| Host      | User       | Password                                     |
+-----+-----+-----+
| localhost | simple_user | *4EF3135F74D7AF2DF378C3A39BC8D1987A3BAC6A |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql> GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX ON `prestashop`
.* TO 'simple_user'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql>
```

Establecimiento de autenticación básica (. htaccess)

Con el fin de proteger mejor su instalación de PrestaShop, es necesario establecer una autenticación básica en el directorio de administración.

Uno de los objetivos del archivo `.htaccess` es proteger sus carpetas y todas sus sub-carpetas. Este sólo funciona en los servidores de Apache, y algunos otros. Asegúrese de que su servidor web es Apache antes de crear un archivo `.htaccess`.

Para lograr la autenticación básica en su carpeta admin, tenemos que agregar un archivo `.htaccess` en esa carpeta (por ejemplo, `/var/www/prestashop/admin`):

```
AuthUserFile /var/www/.prestashop_admin
AuthName "Prestashop Admin Access"
AuthType Basic
Require valid-user
Options -Indexes
```

### Explicación:

- **AuthUserFile:** Muestra la ruta al archivo que contiene los usuarios permitidos y sus contraseñas. `.prestashop_admin` es un archivo de texto.
- **AuthName:** Define el mensaje para mostrar cuando aparece la ventana de autenticación.
- **AuthType:** Define el tipo de autenticación.
- **Require:** Requiere que los usuarios inicien sesión para acceder a los contenidos. `valid-user` permite a varios usuarios a conectarse y acceder la carpeta.
- **Options:** Define las opciones de la carpeta. `-Indexes` deshabilita la generación automática de un índice del directorio si ningún archivo de índice está disponible.

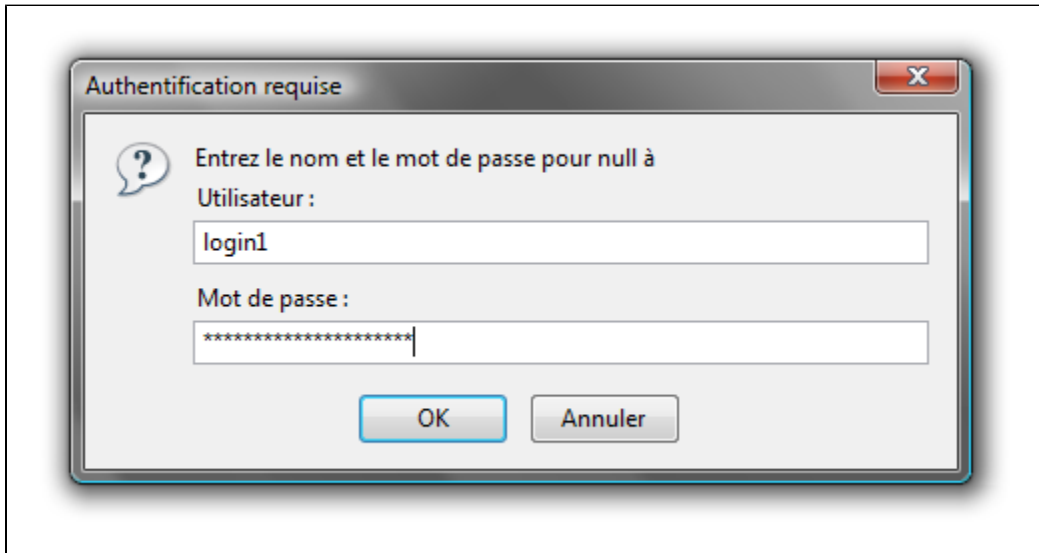
Aquí le ofrecemos una muestra de contenido para el archivo `.prestashop_admin`, con un login y una contraseña:

```
login1:$apr1$/wJeliK8$e9OzgRaVL8J8wSsFBXjor1
login2:$apr1$yV65Kqqz$cFt3sV2.Q7hhLRRUJDo5a/
```

Este archivo contiene logins y contraseñas hash los cuales tienen permitido el acceso a la carpeta. Para generar una contraseña hash, puede seguir este enlace: [generación de archivo .htpasswd](#).

Es muy recomendable poner este archivo en un directorio que sea inaccesible a sus aplicaciones web, en la carpeta `/openbase_dir`. Previene inyección de archivos `.htpasswd`, en caso que una de sus aplicaciones web sea vulnerable.

Ejemplo:



También es posible llevar a cabo restricciones de dominio e IP utilizando el archivo `.htaccess`:

```
Order Allow, Deny
Deny from all
Allow from .myprestashop.com
Allow from 127.0.0.1
```

Sin embargo, usted no debe ingresar este tipo de directiva:

```
<LIMIT GET POST>
Require valid-user
</LIMIT>
```

## Hacer su instalación de PrestaShop más segura

Las siguientes recomendaciones se encuentran ordenadas por orden de importancia:

1. Asegure su back-office
  - a. Cambie el nombre de la carpeta `/admin` después de la instalación de PrestaShop. Esto es necesario ya que usted no podrá acceder a su administración PrestaShop si no ha realizado ese cambio. Asegúrese de escoger un nombre muy singular, lo ideal sería una mezcla de letras y números, como "my4dm1n".
  - b. Proteja su carpeta admin con los archivos `.htaccess` y `.htpasswd`, o solicite a su host que lo haga por usted.
  - c. No permita que su navegador mantenga rastro de su contraseña (cookie o cualquier otro).
  - d. Elija una contraseña compleja, mezclando letras, números e incluso signos de puntuación, como "5r3XaDR#". Usted puede y debe usar un generador de contraseñas, tales como el de [PCTools](#) o el de [GRC](#).
2. Asegure su instalación PHP
  - a. Vea los ajustes necesarios y recomendados de PHP, al principio de esta guía.
3. Siempre elimine la carpeta `/install` después de haber instalado o actualizado PrestaShop.
4. Siempre elimine archivos inútiles del servidor de producción:
  - a. todos los archivos `readme_xx.txt`.
  - b. el archivo `CHANGELOG`.
  - c. la carpeta `/docs`.
5. Prohibir el acceso a los archivos/plantillas de su tema, mediante un archivo `.htaccess` con el siguiente contenido:

```
<FilesMatch "\.tpl$" >
order deny,allow
deny from all
</FilesMatch >
```

## Afinamiento y rendimiento

Esta sección le ayudará a entender mejor las variables de configuración que no se manejan con el back-office, sino directamente en los archivos de configuración.

Hay cuatro archivos de configuración en PrestaShop, todos en la carpeta `/config`:

- `config.inc.php`: archivo central de configuración de PrestaShop.
- `defines.inc.php`: contiene todos los valores constantes de PrestaShop. `__` Previamente definidos en `__settings.inc.php`.
- `settings.inc.php`: contiene la información de acceso a la base de datos, así como el número de versión de PrestaShop.
- `smarty.config.inc.php`: contiene toda la configuración relativa a Smarty, el motor de plantillas/tema utilizado por PrestaShop.

### archivo `config.inc.php`

En modo de producción:

- asegúrese de mantener `@ini_set('display_errors', 'Off');` en "Off".
- asegúrese de mantener `define('PS_DEBUG_SQL', false);` en "false".

En cambio, en el modo de desarrollo/prueba, puede recibir ayuda de seguimiento de posibles errores al:

- cambiar `@ini_set('display_errors', 'Off');` por "On".
- cambiar `define('PS_DEBUG_SQL', false);` por "true".

### archivo `defines.inc.php`

Entre otros valores constantes, este archivo contiene la ubicación de todos los archivos y carpetas. Si usted necesita cambiarlas, no olvide de mantener el original a la mano, en caso de que desee volver al camino original.

### archivo `smarty.inc.php`

- `$smarty->caching = false`: el sistema de caché de Smarty debe estar desactivado, ya que no es compatible con PrestaShop.
- **IMPORTANTE**: en el modo de producción `$smarty->force_compile` se debe establecer en "false", ya que ofrecerá una mejora del 30% en el tiempo de carga de página. Por otro lado, cuando se edita un archivo `.tpl`, tendrá que eliminar el contenido de la carpeta `/tools/smarty/compile` (excepto `index.php`) para observar los cambios en vivo. Note que esta opción también se puede realizar en el back-office, en la opción sub-pestaña "Preferencias">"Performance", en la sección "Smarty".
- `$smarty->compile_check` se debe mantener en "false".
- `$smarty->debugging` le ofrece acceso a la información de depuración de Smarty cuando sus páginas son mostradas.

## Mejoramiento del rendimiento de PrestaShop

Aquí le ofrecemos unos cuantos consejos que deberían permitirle optimizar a PrestaShop.

- Habilitar caché de MySQL (o pregunte a su proveedor de host que lo haga por usted), y ofréczcale un valor alto (por ejemplo, 256M).
- No se olvide de colocar `$smarty->force_compile` en "false" en el modo de producción, ya sea a través del archivo `smarty.inc.php` o en el back-office.
- Siempre que sea posible, utilice un opcóodigo de caché (o pregunte a su host para instalar uno), para aliviar la carga procesada por el servidor. PrestaShop es compatible con [eAccelerator](#). Opcóodigo significa "código de operación" y define el estado compilado de los archivos dinámicos, los que pueden procesar más rápido.
- Si es posible, divida sus elementos estáticos entre diferentes dominios y subdominios, para obtener conexiones paralelas HTTP. Para colocar esto en su lugar, abra el archivo `/config/defines.inc.php` agregue estas líneas (adaptadas a sus necesidades):

```

if ( $_SERVER['REMOTE_ADDR'] != '127.0.0.1' )
{
    define( '_THEME_IMG_DIR_',      'http://img2.xxx.com/'      );
    define( '_THEME_CSS_DIR_',      'http://css.xxx.com/'      );
    define( '_THEME_JS_DIR_',       'http://js.xxx.com/'       );
    define( '_THEME_CAT_DIR_',      'http://img1.xxx.com/c/'   );
    define( '_THEME_PROD_DIR_',     'http://img1.xxx.com/p/'   );
    define( '_THEME_MANU_DIR_',     'http://img1.xxx.com/m/'   );
    define( '_PS_IMG_',             'http://img1.xxx.com/'     );
    define( '_PS_ADMIN_IMG_',       'http://img1.xxx.com/admin/' );
} else {
    define( '_THEME_IMG_DIR_',      __THEMES_DIR__ . __THEME_NAME__ . '/img/' );
    define( '_THEME_CSS_DIR_',      __THEMES_DIR__ . __THEME_NAME__ . '/css/' );
    define( '_THEME_JS_DIR_',       __THEMES_DIR__ . __THEME_NAME__ . '/js/' );
    define( '_THEME_CAT_DIR_',      __PS_BASE_URI__ . 'img/c/' );
    define( '_THEME_PROD_DIR_',     __PS_BASE_URI__ . 'img/p/' );
    define( '_THEME_MANU_DIR_',     __PS_BASE_URI__ . 'img/m/' );
    define( '_PS_IMG_',             __PS_BASE_URI__ . 'img/' );
    define( '_PS_ADMIN_IMG_',       __PS_IMG__ . 'admin/' );
}

```

## Otras recomendaciones

### Modo seguro

**El modo seguro de PHP se encuentra obsoleto en la última versión de PHP y no debe ser utilizado.** Para PrestaShop en particular, tener Modo Seguro activado puede hacer que los módulos de pago se vuelvan inservibles.

### Actualizaciones

El código de sus aplicaciones PHP es el único camino vulnerable a su servidor. Por lo tanto, es muy recomendable actualizar siempre sus aplicaciones de servidor: PHP, MySQL, Apache y cualquier otra aplicación en el cual se ejecuta su sitio.

## Misceláneo

### Estructura de archivos PrestaShop

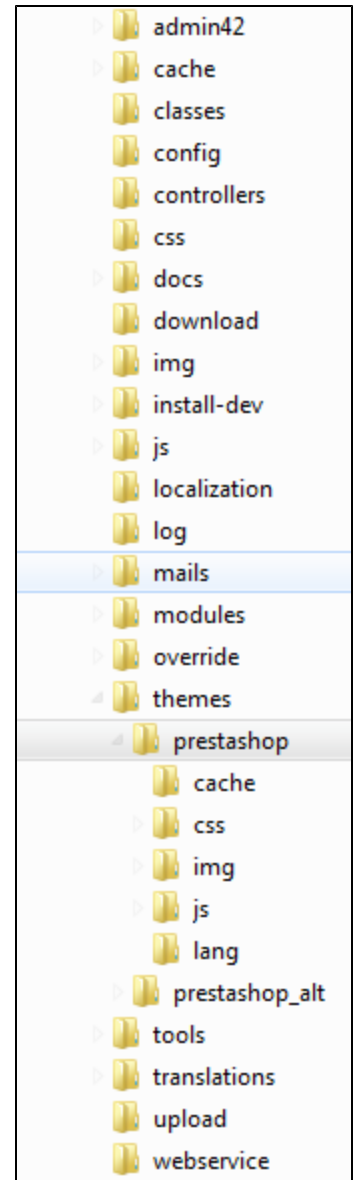
Los desarrolladores de PrestaShop han hecho todo lo posible para separar de manera clara e intuitiva las diferentes partes del software.

Así es como se organizan los archivos:

- /admin: contiene todos los archivos de PrestaShop relacionados con el back-office. Al acceder a esta carpeta con su navegador, se le pedirá que proporcione identificación adecuada, por razones de seguridad. **Importante:** ¡debe asegurarse de proteger esa carpeta con un archivo .htaccess o .htpasswd!
- /cache: contiene las carpetas temporales que son generadas y se vuelven a utilizar para aliviar la carga del servidor.
- /classes: contiene todos los archivos que pertenecen al modelo objeto de PrestaShop. Cada archivo representa (y contiene) una clase de PHP, y sus métodos/propiedades.
- /config: contiene todos los archivos de configuración de PrestaShop. A menos que se lo soliciten, nunca los debe editar, ya que están directamente manejados por el instalador de PrestaShop y el back-office.
- /controllers: contiene todos los archivos relacionados con los controladores de PrestaShop - como en el Model View Controller (o MVC), la arquitectura de software utilizada por PrestaShop. Cada archivo controla una parte específica de PrestaShop.
- /css: contiene todos los archivos CSS que no están vinculados a los temas - por lo tanto, estos son utilizados principalmente por el back-office de PrestaShop.
- /docs: contiene algo de documentación. **Nota:** debe ser eliminada en un entorno de producción.
- /download: contiene sus productos digitales, los cuales pueden ser descargados: PDF, MP3, etc.



- `/img`: contiene todas las imágenes predeterminadas de PrestaShop, iconos y archivos de imagen - aquellos que no pertenecen al tema. Aquí es donde puede encontrar las imágenes por categorías de productos (sub-carpeta `/c`), aquellas de los productos (sub-carpeta `/p`) y aquellas de la propia back-office (sub-carpeta `/admin`).
- `/install`: contiene todos los archivos relacionados con la instalación de PrestaShop. Usted los tendrá que eliminar después de la instalación, para aumentar la seguridad.
- `/js`: contiene todos los archivos de JavaScript que no están vinculados a los temas. La mayoría de ellos pertenecen al back-office. Aquí se encuentra el marco jQuery.
- `/localization`: contiene todos los archivos de localización PrestaShop - es decir, archivos que contengan información local, como divisa, idioma, reglas de impuesto y grupos de reglas de impuestos, estados y las distintas unidades utilizadas en el país elegido (p.ej. el volumen en litros, el peso en kilogramos, etc.)
- `/log`: contiene los archivos de registro generados por PrestaShop en varias etapas, por ejemplo, durante el proceso de instalación.
- `/mails`: contiene todos los archivos HTML y de texto relacionados con correos electrónicos enviados por PrestaShop. Cada idioma tiene su carpeta específica, donde puede editar su contenido manualmente si así lo desea.
- `/modules`: contiene todos los módulos de PrestaShop, cada uno en su propia carpeta. Si desea eliminar definitivamente un módulo, primero desinstálelo desde el back-office, sólo luego de este paso puede eliminar su carpeta.
- `/override`: este es uno especial que apareció con PrestaShop 1.4. Mediante el uso regular de convención de carpeta/nombre de archivo de PrestaShop, es posible crear archivos que anulen los controladores y las clases por defecto de PrestaShop. Esto le permite cambiar el comportamiento del núcleo de PrestaShop, sin modificar los archivos originales, manteniéndolos a salvo para la próxima actualización.
- `/themes`: contiene todos los temas instalados actualmente, cada uno en su propia carpeta.
- `/tools`: contiene herramientas externas que fueron integradas en PrestaShop. Por ejemplo, aquí encontrará Smarty (motor de plantilla/tema), FPDF (generador de archivo PDF), Swift (remite de correo electrónico), PEAR XML Parser (herramienta de PHP).
- `/translations`: contiene una subcarpeta para cada idioma disponible. Sin embargo, si desea cambiar la traducción, debe hacerlo a través de la herramienta interna de PrestaShop, y no editarlos directamente en esta carpeta.
- `/upload`: contiene los archivos que pueden ser cargados por los clientes para productos personalizados (por ejemplo, una imagen que un cliente quiere imprimir en una taza).
- `/webservice`: contiene los archivos que permiten a las aplicaciones de terceros acceder a PrestaShop a través de su API.



## Traslado de PrestaShop

Una instalación de PrestaShop pocas veces permanece en el mismo lugar físico. Existen muchas razones por las que puede necesitar mover sus archivos y datos de PrestaShop:

- Trasladar su tienda desde su equipo local al servidor en línea.
- Trasladar su tienda desde un sub-dominio de prueba al dominio principal.
- Trasladar su tienda de un servidor a otro.
- Trasladar su tienda de un nombre de dominio a otro.

En todas estas circunstancias, usted debe tener cuidado en moverlas adecuadamente, tanto sus archivos (incluyendo imágenes y temas personalizados, y los módulos que haya comprado ...) así como todos sus datos (que se encuentran en su base de datos MySQL).

## Traslado de PrestaShop a un nuevo servidor

Estos son los pasos principales al cambiar de servidor o copiar de su disco duro local al servidor en línea:

1. Ponga su tienda en el modo de mantenimiento, a fin de no perder clientes nuevos o pedidos mientras se mueven los datos. Diríjase a su back-office y bajo la pestaña "Preferencias", establezca la opción "Activar tienda" a "No".
2. Mueva sus archivos
  - a. **Realice una copia de seguridad de todos los archivos:** conecte a su servidor FTP y copie todos los archivos y carpetas en su disco duro local.
  - b. **Transfiera sus archivos a su nuevo host:** conecte con el servidor FTP para su nuevo host y copie todos los archivos y carpetas que acaba de descargar a su disco duro local, así como se encuentran.
3. Mueva sus datos
  - a. **Haga una copia de seguridad de su base de datos ("dump"):** conecte a phpMyAdmin y haga clic en la pestaña "Exportar", seleccione la base de datos de la instalación de PrestaShop y haga clic en el botón "Continuar". Guarde el archivo descargado en su disco duro. Si el tiempo de espera de phpMyAdmin termina antes que pueda exportar todos sus datos, contacte con su host.
  - b. **Transfiera el SQL dump a la nueva base de datos:** conecte al nuevo servidor de phpMyAdmin y haga clic en la opción "Importar", haga clic en el botón "Examinar ..." y busque el archivo SQL que acaba de descargar, luego hacer clic en el botón "Continuar" para cargarlo. Si el tiempo de espera de phpMyAdmin termina antes de que pueda importar todos sus datos, contacte con su nuevo host.
4. Configuración

- a. En el nuevo servidor, abra el archivo /config/settings.inc.php y actualice las configuraciones del nuevo servidor de base de datos (con sus propias configuraciones en lugar de los ejemplos aquí mostrados):
    - define('DB\_SERVER', 'sql.domainname.com');
    - define('DB\_NAME', 'prestashop');
    - define('DB\_USER', 'PS-user');
    - define('DB\_PASSWD', 'djsf15');
    - define('DB\_PREFIX', 'ps\_');
  - b. (1,4 y anteriores) En ese mismo archivo, actualice la configuración de Base URI (siendo '/' el servidor raíz):
    - define('\_PS\_BASE\_URI\_', '/prestashop/');
  - c. Inicie sesión en su Back Office y luego diríjase a la pestaña "Preferencias", seleccione la sub-pestaña "SEO y URLs", y cambie el nombre de dominio al nuevo dominio. Haga lo mismo para el dominio SSL. Esto actualizará las hileras "PS\_SHOP\_DOMAIN" y "PS\_SHOP\_DOMAIN\_SSL" en la tabla de SQL "ps\_configuration".
  - d. En su back-office, diríjase a la pestaña "Herramientas", sub-pestaña "Generadores", y regenere los archivos .htaccess y robots.txt.
5. Conecte a su servidor FTP nuevo y elimine todo excepto los archivos index.php en las siguientes carpetas:
    - /tools/smarty/cache
    - /tools/smarty/compile
    - /tools/smarty\_v2/cache
    - /tools/smarty\_v2/compile
  6. Luego, diríjase a su back-office, y en la pestaña "Preferencias", establezca la opción "Activar tienda" a "Sí".

¡Usted debiera estar ya listo! Compruebe que todos los enlaces se encuentren funcionando y que todos sus productos, imágenes, módulos y temas continúan ahí, trate de crear una nueva cuenta y realizar un pedido con el fin de asegurarse que su tienda esté funcionando como lo esperaba.

### Traslado de PrestaShop a un nuevo dominio

Estos son los pasos principales a la hora que de trasladar PrestaShop a un nuevo dominio en el mismo servidor. Se trata principalmente de una versión más simple de los pasos anteriores - sin tocar los datos, los cuales se mantienen en el mismo servidor MySQL.

1. Ponga su tienda en el modo de mantenimiento, a fin de no perder nuevos clientes o pedidos al mover datos. Diríjase al back-office, y bajo la pestaña "Preferencia", establezca la opción "Activar tienda" a "No".
2. Mueva sus archivos
  - a. **Realice una copia de seguridad de todos los archivos:** conecte a su servidor FTP y copie todos los archivos y carpetas en su disco duro local.
  - b. **\*Transfiera sus archivos a su nuevo host:** conecte con el servidor FTP para su nuevo host y copie todos los archivos y carpetas que acaba de descargar a su disco duro local, así como se encuentran.
3. Configuración
  - a. En el nuevo servidor, abra el archivo /config/settings.inc.php y actualice las configuraciones del nuevo servidor de base de datos (con sus propias configuraciones en lugar de los ejemplos aquí mostrados):
    - define('DB\_SERVER', 'sql.domainname.com');
    - define('DB\_NAME', 'prestashop');
    - define('DB\_USER', 'PS-user');
    - define('DB\_PASSWD', 'djsf15');
    - define('DB\_PREFIX', 'ps\_');
  - b. (1,4 y anteriores) En ese mismo archivo, actualice la configuración de Base URI (siendo '/' el servidor raíz):
    - define('\_PS\_BASE\_URI\_', '/prestashop/');
  - c. Inicie sesión en su Back Office y luego diríjase a la pestaña "Preferencias", seleccione la sub-pestaña "SEO y URLs", y cambie el nombre de dominio al nuevo dominio. Haga lo mismo para el dominio SSL. Esto actualizará las hileras "PS\_SHOP\_DOMAIN" y "PS\_SHOP\_DOMAIN\_SSL" en la tabla de SQL "ps\_configuration".
  - d. En su back-office, diríjase a la pestaña "Herramientas", sub-pestaña "Generadores", y regenere los archivos .htaccess y robots.txt.
4. Conecte a su servidor FTP nuevo y elimine todo excepto los archivos index.php en las siguientes carpetas:
  - /tools/smarty/cache
  - /tools/smarty/compile
  - /tools/smarty\_v2/cache
  - /tools/smarty\_v2/compile
5. Luego, diríjase a su back-office, y en la pestaña "Preferencias", establezca la opción "Activar tienda" a "Sí".

¡Usted debiera estar ya listo! Compruebe que todos los enlaces se encuentren funcionando y que todos sus productos, imágenes, módulos y temas continúan ahí, trate de crear una nueva cuenta y realizar un pedido con el fin de asegurarse que su tienda esté funcionando como lo esperaba.