

2016

# IAW\_U2\_T1

## Configurar Wordpress

ALEJANDRO VALCÁRCEL MAQUEDA

Práctica Nº2 de Implantación de Aplicaciones Web, para el módulo de administración de sistemas informáticos en red. Se instala y configura Wordpress sobre un servidor apache y MySQL.



## Tabla de contenido

1. Versiones .....	3
2. Tablas que Wordpress crea de manera automática .....	3
3. Crear la base de datos Diagram .....	4
4. Encontrar el archivo de configuración y explicar cada opción. ....	6
5. Estructura de archivos .....	8
6. Proceso de añadir un plugin .....	11
7. Proceso de añadir un Tema .....	12
8. Diez medidas de seguridad para Wordpress .....	13



## 1. Versiones

**Sistema operativo:** Windows 10 x64 ver1607  
**Apache:** ver 2.4  
**PHP:** ver 5.6.24  
**MySQL:** ver 15.1  
**Wordpress:** 4.6.1

## 2. Tablas que Wordpress crea de manera automática

Tabla	Acción	Filas	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> wp_commentmeta	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_unicode_ci	48 KB	-
<input type="checkbox"/> wp_comments	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	1	InnoDB	utf8mb4_unicode_ci	96 KB	-
<input type="checkbox"/> wp_links	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_unicode_ci	32 KB	-
<input type="checkbox"/> wp_options	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	128	InnoDB	utf8mb4_unicode_ci	1.5 MB	-
<input type="checkbox"/> wp_postmeta	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	1	InnoDB	utf8mb4_unicode_ci	48 KB	-
<input type="checkbox"/> wp_posts	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	3	InnoDB	utf8mb4_unicode_ci	80 KB	-
<input type="checkbox"/> wp_termmeta	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	0	InnoDB	utf8mb4_unicode_ci	48 KB	-
<input type="checkbox"/> wp_terms	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	1	InnoDB	utf8mb4_unicode_ci	48 KB	-
<input type="checkbox"/> wp_term_relationships	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	1	InnoDB	utf8mb4_unicode_ci	32 KB	-
<input type="checkbox"/> wp_term_taxonomy	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	1	InnoDB	utf8mb4_unicode_ci	48 KB	-
<input type="checkbox"/> wp_usermeta	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	15	InnoDB	utf8mb4_unicode_ci	48 KB	-
<input type="checkbox"/> wp_users	★ Examinar Estructura Buscar Insertar Vaciar Eliminar	1	InnoDB	utf8mb4_unicode_ci	64 KB	-
12 tablas	Número de filas	152	InnoDB	utf8_spanish_ci	2.1 MB	0 B

**Wp\_commentmeta:** cada comentario que se haga incluye sus propios datos metadata, y estos se almacenan en esta tabla.

**Wp\_comments:** comentarios que se realicen en el Wordpress

**Wp\_links:** se almacenan la información sobre los enlaces añadidos con la funcionalidad propia que tiene Wordpress para ello

**Wp\_options:** se almacenan los ajustes generales sobre la administración del sistema, como usuario, nombre, email, etc

**Wp\_postmeta:** cada tabla que se añade contiene su propia información llamada data, la cual se almacena en esta tabla.

**Wp\_posts:** aquí se almacenan las entradas, los elementos del menú, y las páginas que añadamos.

**Wp\_termmeta:** permite añadir metadatos a los terms.

**Wp\_terms:** almacena las categorías de las entradas y de los enlaces, así como las etiquetas de las entradas

**Wp\_terms\_relationships:** las entradas y enlaces se relacionan con etiquetas/categorías de la tabla wp\_terms. Esta relación se almacena aquí.

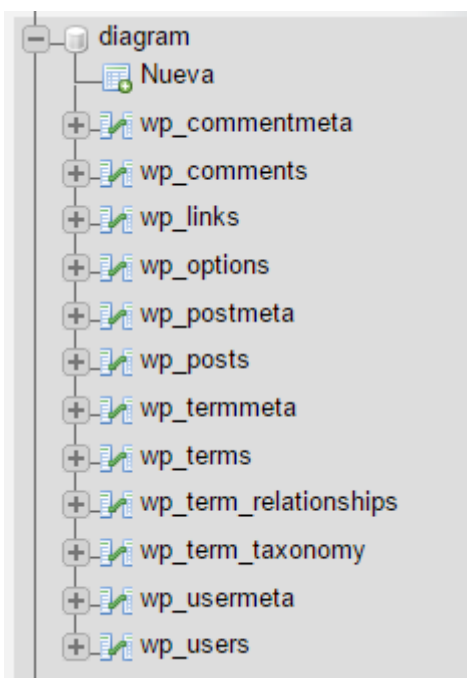
**Wp\_term\_taxonomy:** esta tabla describe la taxonomía (categoria, enlace, o etiqueta) de cada línea de las wp\_terms.

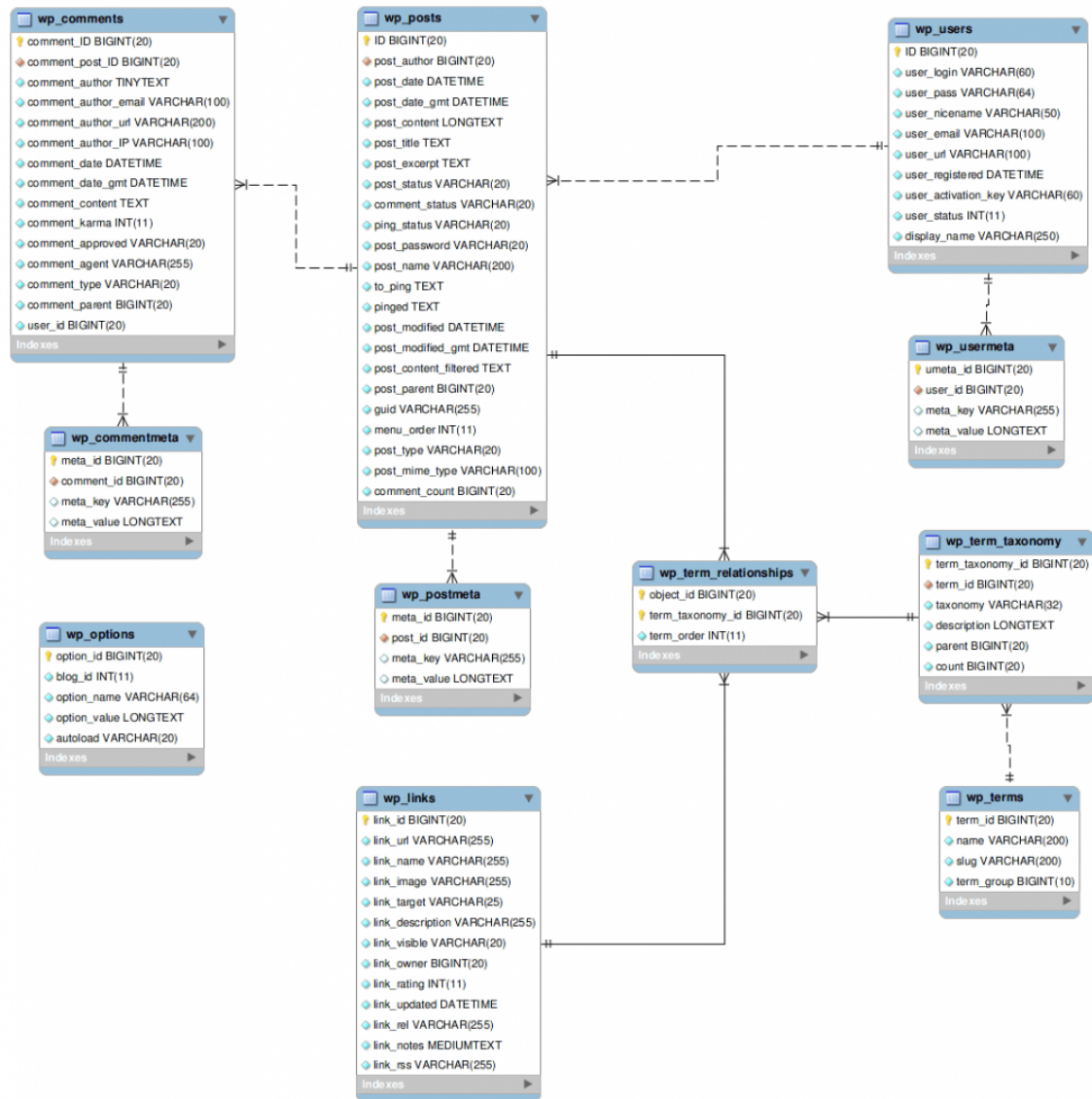
**Wp\_usermeta:** almacena la información data de cada usuario

**Wp\_users:** almacena la lista de usuarios existente.

### 3. Crear la base de datos Diagram

Pulsando sobre nueva, podemos añadir una nueva base de datos. Yo la he añadido previamente a la instalación de Wordpress. El resultado es el siguiente:





## 4. Encontrar el archivo de configuración y explicar cada opción.

El archivo se encuentra en C://xampp/htdocs/diagram/wp\_config.php. Muestra el siguiente contenido:

```
17 // ** Ajustes de MySQL. Solicita estos datos a tu proveedor de alojamiento web. ** /
18 /** El nombre de tu base de datos de WordPress */
19 define('DB_NAME', 'diagram');
20
21 /** Tu nombre de usuario de MySQL */
22 define('DB_USER', 'root');
23
24 /** Tu contraseña de MySQL */
25 define('DB_PASSWORD', '201296');
26
27 /** Host de MySQL (es muy probable que no necesites cambiarlo) */
28 define('DB_HOST', 'localhost');
29
30 /** Codificación de caracteres para la base de datos. */
31 define('DB_CHARSET', 'utf8mb4');
32
33 /** Cotejamiento de la base de datos. No lo modifiques si tienes dudas. */
34 define('DB_COLLATE', '');
```

En estos primeros ajustes encontramos:

- DB\_NAME: nombre de la base de datos
- DB\_USER: usuario que se una para la conexión a la base de datos
- DB\_PASSWORD: contraseña del usuario que se usa para conectarse a la base de datos
- DB\_HOST: host de MySQL que usamos con la base de datos
- DB\_CHARSET: codificación de caracteres que usa la base de datos
- DB\_COLLATE: cotejamiento de la base de datos (orden del conjunto de caracteres)

```
* @since 2.6.0
*/
define('AUTH_KEY', 'Y09_%-pbx^v1^|hDixuw^T8~0P;nD8(?Ir1^@?/}QP*X `NEya~]Cl&.C5EbG2[l');
define('SECURE_AUTH_KEY', '5<03^|22Gq<0o@$*z|j2/e(~0ntXI,Q*UQ(Dtxu>QQX9.5BX4!%}jVgEj1=Vbxo+');
define('LOGGED_IN_KEY', 'Y]=oMrFsM8c[{X34*h3R{^}l$`JwbWn/&uw~ynBR?P_jGmg8GJC^X3-[95!3F{tk}');
define('NONCE_KEY', 'nc@vie)tjrLP/ul6K9[-[l6z;k.PZ]qGZ?!?v-#`]cs<)>#wE-*Zj8<FV-.n&Ut9t');
define('AUTH_SALT', 'SUEb5Qbl#o]U.shOp)edH_YdHmw8ENQQfp$erz`hs^+Ydir=^}ISQ4XryUhar-Nq');
define('SECURE_AUTH_SALT', '/PT[]Z;@BQf|MZ+B).JBrZY4~Sak+0gy:-#`=jg{fQTZ50vb}Q*>tYr!lLq)Q[pS');
define('LOGGED_IN_SALT', '+Nq,<5Z?EZ!|-pAHcEh4Gi>22VbLuGCX%AShDgI@~j2JI(h|pd1lhuxF8!Gs;0pA');
define('NONCE_SALT', 'UJDBx}*|&HbG8p9u42*qj!^t*i)qal(l4c)/PUL{WH<jw@4_<PuZiK@tjwxd!oAT}');

/**#@-*/

/**
 * Prefijo de la base de datos de WordPress.
 *
 * Cambia el prefijo si deseas instalar multiples blogs en una sola base de datos.
 * Emplea solo números, letras y guión bajo.
 */
$table_prefix = 'wp_';
```

A continuación se muestran las siguientes opciones de configuración:

**-AUTH\_KEY, SECURE\_AUTH\_KEY, LOGGED\_IN\_KEY:** claves que proporcionan una mejor encriptación sobre las cookies almacenadas del usuario.

**-NONCE\_KEY:** otra clave como la anterior añadida a partir de la versión 2.7

**-AUTH\_SALT, SECURE\_AUTH\_SALT, LOGGED\_IN\_SALT, NONCE\_SALT:** cada una de las claves anteriores tiene su propio salt, que es recomendado, aunque no obligatorio.

**-TABLE\_PREFIX:** pertenece ya a las opciones de configuración avanzada. Es el valor de comienzo de los nombres de las tablas de las bases de datos. En este caso “wp\_”.

Lo siguiente ya pertenece todo a la configuración avanzada:

```
/**
 * Para desarrolladores: modo debug de WordPress.
 *
 * Cambia esto a true para activar la muestra de avisos durante el desarrollo.
 * Se recomienda encarecidamente a los desarrolladores de temas y plugins que usen WP_DEBUG
 * en sus entornos de desarrollo.
 */
define('WP_DEBUG', false);

/* ¡Eso es todo, deja de editar! Feliz blogging */

/** WordPress absolute path to the Wordpress directory. */
if ( !defined('ABSPATH') )
    define('ABSPATH', dirname(__FILE__) . '/');

/** Sets up WordPress vars and included files. */
require_once(ABSPATH . 'wp-settings.php');
```

**-WP\_DEBUG:** controla el reporte de errores y advertencias que se puedan producir durante el uso. La línea que lo continua, es la ruta donde se almacenarán.



## 5. Estructura de archivos

wp-admin	16/09/2016 10:09	Carpeta de archivos	
wp-content	27/09/2016 9:53	Carpeta de archivos	
wp-includes	16/09/2016 10:13	Carpeta de archivos	
.htaccess	27/09/2016 9:53	Archivo HTACCESS	1 KB
index	25/09/2013 0:18	Archivo PHP	1 KB
licencia	16/09/2016 10:13	Documento de tex...	18 KB
license	16/09/2016 10:12	Documento de tex...	20 KB
readme	16/09/2016 10:13	Chrome HTML Do...	8 KB
wp-activate	24/05/2016 21:02	Archivo PHP	6 KB
wp-blog-header	19/12/2015 10:20	Archivo PHP	1 KB
wp-comments-post	23/05/2016 16:44	Archivo PHP	2 KB
wp-config	27/09/2016 9:46	Archivo PHP	4 KB
wp-config-sample	16/09/2016 10:13	Archivo PHP	4 KB
wp-cron	24/05/2015 17:26	Archivo PHP	4 KB
wp-links-opml	23/05/2016 16:44	Archivo PHP	3 KB
wp-load	14/04/2016 17:53	Archivo PHP	4 KB
wp-login	14/06/2016 21:51	Archivo PHP	34 KB
wp-mail	13/07/2016 12:37	Archivo PHP	8 KB
wp-settings	13/08/2016 16:02	Archivo PHP	14 KB
wp-signup	24/05/2016 20:44	Archivo PHP	30 KB
wp-trackback	30/11/2014 20:23	Archivo PHP	4 KB
xmlrpc	06/07/2016 12:40	Archivo PHP	3 KB

Este es el contenido de la carpeta principal Diagram. Encontramos tres carpetas, además de los archivos de configuración general, como el wp-config. A continuación se muestra para que sirve cada carpeta.

e equipo > Disco local (C:) > xampp > htdocs > Diagram > wp-admin				
Nombre	Fecha de modifica...	Tipo	Tamaño	
css	16/09/2016 10:09	Carpeta de archivos		
images	16/09/2016 10:09	Carpeta de archivos		
includes	16/09/2016 10:09	Carpeta de archivos		
js	16/09/2016 10:09	Carpeta de archivos		
maint	16/09/2016 10:09	Carpeta de archivos		
network	16/09/2016 10:09	Carpeta de archivos		
user	16/09/2016 10:09	Carpeta de archivos		
about	07/09/2016 14:57	Archivo PHP	13 KB	
admin	08/07/2016 14:37	Archivo PHP	11 KB	
admin-ajax	26/07/2016 23:24	Archivo PHP	4 KB	
admin-footer	09/07/2016 13:23	Archivo PHP	3 KB	
admin-functions	06/07/2016 12:40	Archivo PHP	1 KB	
admin-header	25/05/2016 16:05	Archivo PHP	7 KB	
admin-post	25/02/2016 11:53	Archivo PHP	2 KB	
async-upload	29/06/2016 15:16	Archivo PHP	5 KB	
comment	29/06/2016 15:16	Archivo PHP	11 KB	
credits	10/08/2016 23:57	Archivo PHP	5 KB	
custom-background	10/07/2016 0:51	Archivo PHP	17 KB	
custom-header	29/06/2016 15:16	Archivo PHP	44 KB	
customize	29/06/2016 15:16	Archivo PHP	7 KB	
edit	17/07/2016 16:05	Archivo PHP	16 KB	
edit-comments	29/06/2016 15:16	Archivo PHP	13 KB	
edit-form-advanced	10/07/2016 0:56	Archivo PHP	32 KB	
edit-form-comment	10/06/2016 4:50	Archivo PHP	8 KB	
edit-link-form	14/10/2015 17:32	Archivo PHP	6 KB	
edit-tag-form	14/06/2016 21:51	Archivo PHP	9 KB	
edit-tags	17/07/2016 16:15	Archivo PHP	21 KB	
export	29/06/2016 15:16	Archivo PHP	11 KB	
freedoms	10/08/2016 23:57	Archivo PHP	4 KB	



Carpeta **wp-admin**. Contiene todos los archivos principales para el funcionamiento de la página: estructuras principales html, configuraciones, java scripts...etc. Además, contiene varias subcarpetas:

-**css**: contiene las hojas de estilo de la página

-**js**: contiene los archivos de java scripts aplicados a la página

-**maint**: archivo de reparación

-**images**: todas las imágenes que aparecen en la página

-**includes**: librerías y archivos php para el funcionamiento

-**network**: contiene los archivos que hacen funcionar la pantalla de administración de conexiones y generales.

-**user**: contiene la configuración para mantener el listado de usuarios y sus interfaces.

ite equipo > Disco local (C:) > xampp > htdocs > Diagram > wp-content				
Nombre	Fecha de modifica...	Tipo	Tamaño	
languages	16/09/2016 10:13	Carpeta de archivos		
plugins	16/09/2016 10:09	Carpeta de archivos		
themes	16/09/2016 10:10	Carpeta de archivos		
index	08/01/2012 16:01	Archivo PHP	1 KB	

Carpeta **wp-content**. En su interior se encuentran tres subcarpetas:

-**languages**: archivos de la configuración del idioma y su conjunto de caracteres establecido, para el funcionamiento de la página y de los temas configurados.

-**plugins**: contiene los plugins instalados en nuestra página de Wordpress.

-**themes**: incluye los temas para la página que tengamos instalados.



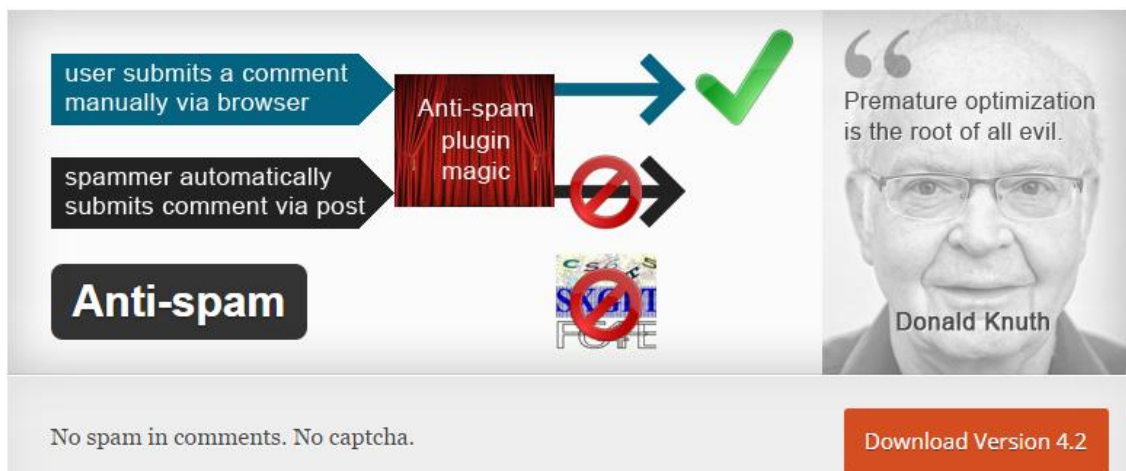
equipo &gt; Disco local (C:) &gt; xampp &gt; htdocs &gt; Diagram &gt; wp-includes

Nombre	Fecha de modifica...	Tipo	Tamaño
certificates	16/09/2016 10:09	Carpeta de archivos	
css	16/09/2016 10:09	Carpeta de archivos	
customize	16/09/2016 10:09	Carpeta de archivos	
fonts	16/09/2016 10:09	Carpeta de archivos	
ID3	16/09/2016 10:09	Carpeta de archivos	
images	16/09/2016 10:09	Carpeta de archivos	
js	16/09/2016 10:09	Carpeta de archivos	
pomo	16/09/2016 10:09	Carpeta de archivos	
random_compat	16/09/2016 10:09	Carpeta de archivos	
Requests	16/09/2016 10:09	Carpeta de archivos	
rest-api	16/09/2016 10:09	Carpeta de archivos	
SimplePie	16/09/2016 10:09	Carpeta de archivos	
Text	16/09/2016 10:09	Carpeta de archivos	
theme-compat	16/09/2016 10:09	Carpeta de archivos	
widgets	16/09/2016 10:09	Carpeta de archivos	
admin-bar	23/05/2016 18:59	Archivo PHP	25 KB
atomlib	19/05/2016 0:06	Archivo PHP	11 KB
author-template	06/07/2016 12:40	Archivo PHP	15 KB
bookmark	27/05/2016 17:16	Archivo PHP	14 KB
bookmark-template	22/05/2016 18:24	Archivo PHP	12 KB
cache	06/07/2016 12:40	Archivo PHP	22 KB
canonical	09/08/2016 0:13	Archivo PHP	27 KB
capabilities	30/06/2016 1:02	Archivo PHP	22 KB
category	06/07/2016 12:40	Archivo PHP	12 KB
category-template	20/07/2016 16:57	Archivo PHP	48 KB
class.wp-dependencies	14/03/2016 21:39	Archivo PHP	13 KB
class.wp-scripts	06/07/2016 12:40	Archivo PHP	15 KB
class.wp-styles	22/05/2016 18:50	Archivo PHP	10 KB
class-feed	20/07/2016 7:33	Archivo PHP	8 KB

Carpeta wp-includes. Incluye todo lo necesario para funcionar a través de la interfaz de WordPress. Es el territorio de WordPress Core, por tanto, es recomendable que nunca se modifique núcleo bajo ninguna circunstancia. Mientras wp-content puede definir plugins y temas, la API de sí mismo y la gran mayoría del núcleo de WordPress se sitúa en esta carpeta. Todas sus funciones estándar y sus dependencias se encuentran en esta carpeta. Se encuentran las APIs, widgets, hojas CSS, texto de temas, etc.



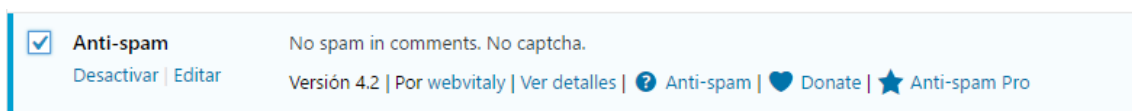
## 6. Proceso de añadir un plugin



Para esta prueba voy a instalar un plugin anti spam, tal y como se ve en la foto anterior. Lo primero es descargarlo, es un archivo ZIP. Una vez extraído, copiamos la carpeta en wp-content/plugins. En la foto se ve el resultado:

te equipo > Disco local (C:) > xampp > htdocs > Diagram > wp-content > plugins				
Nombre	Fecha de modifica...	Tipo	Tamaño	
akismet	16/09/2016 10:09	Carpeta de archivos		
anti-spam	13/03/2016 23:05	Carpeta de archivos		
hello	22/05/2013 21:08	Archivo PHP	3 KB	
index	05/06/2014 15:59	Archivo PHP	1 KB	

Cuando la carpeta ya esté en su lugar, procedemos a activar el plugin. Para ello, vamos al panel de control, y en la ventana de plugins buscamos el que hemos añadido, haciendo click en activar.



## 7. Proceso de añadir un Tema

Desde la página [http://ayudawp.com/downloads/?dl\\_cat=2](http://ayudawp.com/downloads/?dl_cat=2) se pueden descargar varios temas para instalar en el Wordpress. Para ello, primero elegimos uno, lo descargamos, y extraemos el archivo.

### Artificial-intelligence-ayudawp

En esta página encontrarás la **lista de descargas** organizadas por categorías. Puede que algunos archivos no tengan el enlace activo, esto es debido a que algunas descargas requieren que estés registrado en **Ayuda WordPress**. Si quieres acceder a un archivo protegido puedes registrarte [en esta página](#) ...



Download

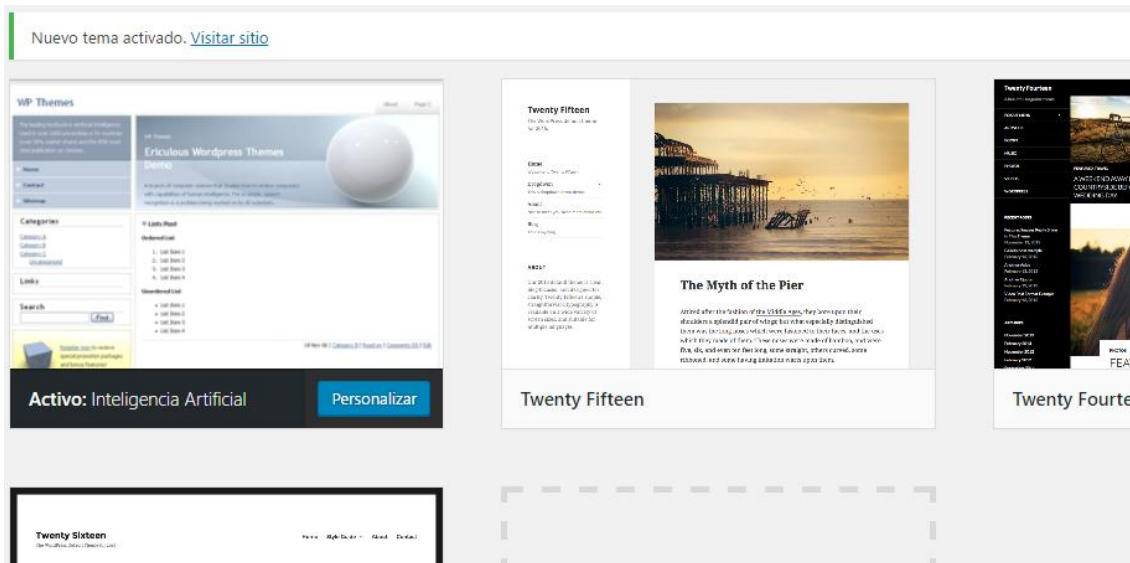
Filename	artificial-intelligence-ayudawp.zip
filesize	79.66 kB
Version	1
Date added	27 marzo, 2010
Downloaded	1115 times
Category	<a href="#">Temas</a>

A continuación, la carpeta extraída del Zip, la cortamos y pegamos en “/wp-content/themes/”, para que Wordpress la detecte.

te equipo > Disco local (C:) > xampp > htdocs > Diagram > wp-content > themes				
Nombre	Fecha de modifica...	Tipo	Tamaño	
artificial-intelligence-10	10/01/2008 17:15	Carpeta de archivos		
twentyfifteen	16/09/2016 10:09	Carpeta de archivos		
twentyfourteen	16/09/2016 10:09	Carpeta de archivos		
twentysixteen	16/09/2016 10:09	Carpeta de archivos		
index	05/06/2014 15:59	Archivo PHP	1 KB	

Por último, accedemos al panel de Administración, y en la pestaña presentación, activamos el nuevo tema añadido.





## 8. Diez medidas de seguridad para Wordpress

Algunas de las medidas de seguridad que más se deben tener en cuenta son:

**1-Limitar el intento de loggins:** Debes dejar uno o dos intentos por si tú mismo te equivocas al intentar entrar al panel de administración de WordPress pero no más. Cualquier hacker que emplee la fuerza bruta para acceder a tu cuenta debe ser bloqueado si prueba más de dos o tres combinaciones de usuario y contraseña erróneas. Existen plugins para controlarlo, como Limit Login Attempts.

**2-Controla el spam:** al tener un sistema de comentarios, el spam es algo muy común. En WordPress existe el plugin Akismet que mantiene a raya el spam se pueden plantear estrategias más drásticas como emplear un sistema de comentarios externo que se encargará por ti de eliminar el spam: Disqus o Google+.

**3-Proteger el directorio wp-admin** con una contraseña usando .htaccess: El archivo .htaccess se puede encontrar en los servidores Apache y pueden servir para desempeñar muchas tareas. Una de ellas sería proteger un determinado directorio de forma que al tratar de acceder a él nos solicite un usuario y contraseña, con lo que añadiremos una capa de seguridad adicional. Para conseguir esto, deberemos crear un archivo de nombre .htaccess (ojo, lleva un punto delante de "htaccess"). Después deberemos editarlo y añadir un código como el siguiente:

```
AuthType Basic
AuthName "Acceso restringido"
AuthUserFile "/home/NOMBRECUENTA/.htpasswd/public_html/wp-admin/passwd"
require valid-user
```

Aquí, /home/NOMBRECUENTA/.htpasswd/public\_html/wp-admin/passwd sería la ruta al archivo que contiene el usuario y contraseña de acceso, en este caso llamado passwd. Es

importante que este archivo se encuentre alojado fuera de la ruta donde se encuentran los archivos de la web, para aumentar la seguridad.

En el archivo passwd deberemos colocar el usuario y contraseña, teniendo en cuenta que esta contraseña va encriptada. Para generar estos datos podemos utilizar una página como la siguiente: <http://www.htaccesstools.com/htpasswd-generator/> Aquí veremos que después de poner el usuario y contraseña nos generará un código. Este código será el que deberemos poner dentro del archivo passwd, que deberá estar colocado en la ruta que hayamos configurado. Para el ejemplo anterior, la ruta sería /home/NOMBRECUESTA/.htpasswd/public\_html/wp-admin

**4-Proteger el archivo wp-login.php.** Para ello podemos agregar una protección de tipo .htaccess, como habíamos visto en la otra entrada para proteger el acceso al directorio wp-admin. En este caso, deberemos editar el archivo .htaccess que tenemos en la raíz de nuestro sitio web, y añadir al comienzo del mismo un código como el siguiente (el archivo ya puede contener otro código que no hay que cambiar):

```
<files wp-login.php>
AuthType Basic
AuthName "Acceso restringido"
AuthUserFile "/home/NOMBRECUESTA/.htpasswd/public_html/passwd"
require valid-user
</files>
```

En el código anterior, /home/NOMBRECUESTA/.htpasswd/public\_html/passwd sería la ruta al archivo que contiene el usuario y contraseña de acceso, en este caso llamado passwd. Es muy recomendable que este archivo se encuentre alojado fuera de la ruta donde se encuentran los archivos de la web, así evitaremos el acceso al mismo vía web.

En el archivo passwd deberemos colocar el usuario y contraseña, teniendo en cuenta que esta contraseña va encriptada. Para generar estos datos podemos utilizar una página como la siguiente: <http://www.htaccesstools.com/htpasswd-generator/>

Aquí veremos que después de poner el usuario y contraseña nos generará un código. Este código será el que deberemos poner dentro del archivo passwd, que deberá estar colocado en la ruta que hayamos configurado. Para el ejemplo anterior, la ruta sería /home/NOMBRECUESTA/.htpasswd/public\_html

De esta forma, cuando accedamos a este archivo para iniciar sesión de usuario de WordPress, nos solicitará un usuario y contraseña adicional previa a la carga de la página.

**5-Ocultar listado de ficheros en tus directorios.** Algunos servidores web están mal configurados, de manera que desde el navegador se puede acceder al index.html o al index.php, obteniendo así toda la estructura de directorios y ficheros que tenemos. Para ocultar esto hacemos el siguiente proceso:

Simplemente añade esto al inicio de tu fichero .htaccess:

```
Options -Indexes
```

Si en lugar de Apache usas Nginx, has de abrir el fichero de configuración de tu VirtualHost y añadir algo así:

```
location / {
    autoindex off;
}
```

Con esta sencilla medida, si alguien intenta listar los directorios de tu web, obtendrá un mensaje 403 Forbidden (Prohibido) .



**6-Ocultar la versión de PHP que se está ejecutando:** Wordpress funciona sobre PHP, de modo que mostrar la versión de este que se está ejecutando sería un error. Si tenemos acceso al archivo `php.ini`, donde se guarda la configuración de PHP, podremos arreglar este problema de forma sencilla, simplemente agregando la línea `expose_php = Off`. En caso de que ya tuviéramos la línea `expose_php = On` bastará con cambiar el valor `On` a `Off`. En caso de que no tengamos acceso a la configuración de PHP, lo más común cuando hemos contratado un servicio de hosting, deberemos solicitar este cambio a los proveedores del servicio. Si esto último tampoco es posible, podemos añadir unas líneas de código en el archivo `.htaccess` de la raíz de la web para evitar que se muestre la versión de PHP:

```
ServerTokens Prod
ServerSignature Off
RewriteCond %{QUERY_STRING} \=PHP[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12} [NC]
RewriteRule .* - [F]
```

**7-Archivos que comprometen la seguridad de Wordpress:** Existen diversos archivos que pueden comprometer la seguridad de WordPress. Por un lado, hay algunos archivos que se añaden con la instalación de WordPress, que son meramente informativos, pero cuya información puede ser útil para los atacantes.

Por ejemplo, podría mostrar información sobre la versión de WordPress que estamos utilizando, algo muy útil si no tenemos la última versión disponible. Por otro lado, existen algunos archivos necesarios para el funcionamiento de WordPress, que no podemos eliminar, y que contienen información sensible. Es conveniente proteger estos archivos.

Empecemos con los primeros. En la raíz de la web solo deberíamos tener archivos con extensión `php` y el archivo `.htaccess`. El resto de archivos, con extensiones `txt` o `html` deberían ser eliminados, a menos que los hayamos añadido nosotros para algún cometido en particular, o se hayan instalado con algún plugin (normalmente ningún plugin debería dejar archivos en la raíz, pero puede que haya excepciones). Los archivos que deberíamos eliminar son los siguientes (puede que no lo tengas todos):

```
-license.txt
-licencia.txt
-readme.html
```

Si, por seguridad, no los quieres eliminar, puedes crear un directorio nuevo donde mover estos archivos o también puedes renombrarlos. Por ejemplo, puedes cambiar el nombre `license.txt` por `license_oculto.txt`. En todo caso, sin estos archivos tu WordPress también funcionará perfectamente.

**8-Proteger el archivo `wp-config.php`.** Una forma de proceder sería agregar un código en el archivo `htaccess` que también encontraremos en la raíz de nuestro WordPress. En concreto habrá que añadir al final del archivo `.htaccess` (este ya debería contener un código) lo siguiente y guardar los cambios:

```
<files wp-config.php>
order allow,deny
deny from all
</files>
```

Con esta modificación, denegaremos el acceso al archivo `wp-config.php` a todos los usuarios, por lo que este archivo solo será accesible para el propio WordPress, que es justo lo que necesitamos.



**9-Cambiar el usuario administrador:** Otro aspecto que quizá está vinculado con los ítems que hemos mencionado anteriormente es éste; una vez que se llega a abrir una nueva cuenta de administrador para un blog de WordPress, las credenciales suelen configurarse automáticamente con un nombre de usuario y una contraseña, siendo esta última la que debemos cambiar en primera instancia por otra más segura. El primer aspecto, es decir, el nombre de usuario generalmente tiene las palabras “admin”, siendo necesario que cambiemos dicho nombre o que lleguemos a crear una nueva cuenta con privilegios de administrador, pero con un nombre diferente.

**10-Plugins con vulnerabilidades:** Si bien es cierto que los plugins pueden ser una gran ayuda para optimizar funciones que no sabemos manejar bien y de forma nativa, mientras menos complementos tengamos, mejor será para nosotros. Trata de explorar a todos los plugins que tengas instalados en tu blog de WordPress, debiendo tratar de desactivar o quizá eliminar a todos aquellos que ya no usas de las que está seguro, que no vas a usar jamás.

