

IFCD0110 - CONFECCIÓN Y PUBLICACIÓN DE PÁGINAS WEB

MF952_2 – PUBLICACIÓN DE PÁGINAS WEB

1. Backup y Restauración de WordPress con XCloner

Introducción

En todo desarrollo software es importante realizar copias de seguridad o backups del trabajo realizado que nos permita recuperarnos de un fallo, de la índole que sea, o que nos permita volver a un estado anterior por cualquier otra razón.

Un **backup** o **copia de seguridad** será entonces *“una copia de los datos originales que se realiza con el fin de disponer de un medio para recuperarlos en caso de su pérdida”*. Todo proceso de copia de seguridad se complementa con otro denominado **restauración de los datos** (en inglés **restore**), que es la acción de leer y grabar en la ubicación original u otra alternativa los datos requeridos. Por esto último también es muy común la utilización del backup/restauración para el traslado de ubicación de los datos y/o aplicaciones, como llevar del ordenador de desarrollo al servidor de producción en internet, la web diseñada, y por su puesto a la inversa. Por lo que los posibles escenarios en los que utilizamos backup y restauración serían:

- Backup local
- Backup remoto
- Backup local y restaruación local
- Backup local y restaruación remota
- Backup remota y restauración local
- Backup remota y restaruación remota

Proceso de Backup o Copia de Seguridad

Backup Manual

En los sitios web que desarrollamos con WordPress el backup se basa en realizar el backup de los archivos del sitio y el backup de su base de datos. Por lo que realizar un backup de WordPress simplemente serían dos pasos:

1. Copiar los archivos, si estamos haciendo copia de un servidor remoto, lo haremos utilizando el servicio de FTP del hosting. Si lo estamos haciendo en local, simplemente haremos la copia con el administrador de archivos.
2. Copia de la base de datos. Independientemente de que la web la tengamos en local o en remoto esta operación la realizaremos a través de phpmyadmin y exportando la base de datos.
 - a) Seleccionar la opción de método de exportación Personalizado y elegir la base de datos que queremos exportar.

← Servidor: 127.0.0.1

Bases de datos | **SQL** | **Estado actual** | **Cuentas de usuarios** | **Exportar** | **Importar**

Método de exportación:

☐ Rápido - mostrar sólo el mínimo de opciones de configuración

☒ Personalizado - mostrar todas las opciones de configuración posibles

Formato:

SQL

Bases de datos:

Seleccionar todo / Deseleccionar todo

- curso
- deporteshgaldar
- easygarden**
- fiestasantiago
- gymforce
- phpmyadmin
- restaurante
- test
- veranojoven
- waterzone

b) En el apartado de *Salida* configurar por ejemplo la plantilla del nombre de archivo con @SERVER@-drop

Salida:

☐ Renombrar las bases de datos/tablas/columnas exportadas

☒ Guardar salida a un archivo

Plantilla del nombre del archivo: @SERVER@-drop ☒ usar esto para exportaciones futuras

Conjunto de caracteres del archivo: utf-8

Compresión: Ninguna

☐ Exportar bases de datos como archivos separados

☐ Ver salida como texto

Omitir tablas mayores a MB

- c) En *Opciones específicas al formato*, asegurarse de tener marcada la opción estructura y datos, lo que indica que se exportará tanto los datos de la base de datos como su estructura de tablas.

Opciones específicas al formato:

- ☒ Mostrar comentarios (*incluye información como marca temporal de exportación, versión PHP y versión de servidor*)

Comentario de cabecera personalizado adicional (\n divide líneas):

- ☐ Incluye una marca temporal de cuando la base de datos fue creada, modificada por última vez y revisada por última vez
- ☐ Mostrar relaciones de clave foránea
- ☐ Tipos MIME disponibles
- ☐ Incluir lo exportado en una transacción
- ☐ Deshabilitar la revisión de las claves foráneas
- ☐ Exportar vistas como tablas
- ☐ Exportar metadatos

Sistema de base de datos o versión antigua de servidor MySQL con la que maximizar la compatibilidad de la salida:

NONE



- ☐ estructura
- ☐ datos
- ☒ estructura y datos

- d) En *Opciones de creación de objetos*, marcar la opción Agregar sentencia DROP TABLE, ya que esto permitirá añadir los datos sin que se produzcan errores por índices duplicados en los campos clave de las tablas.

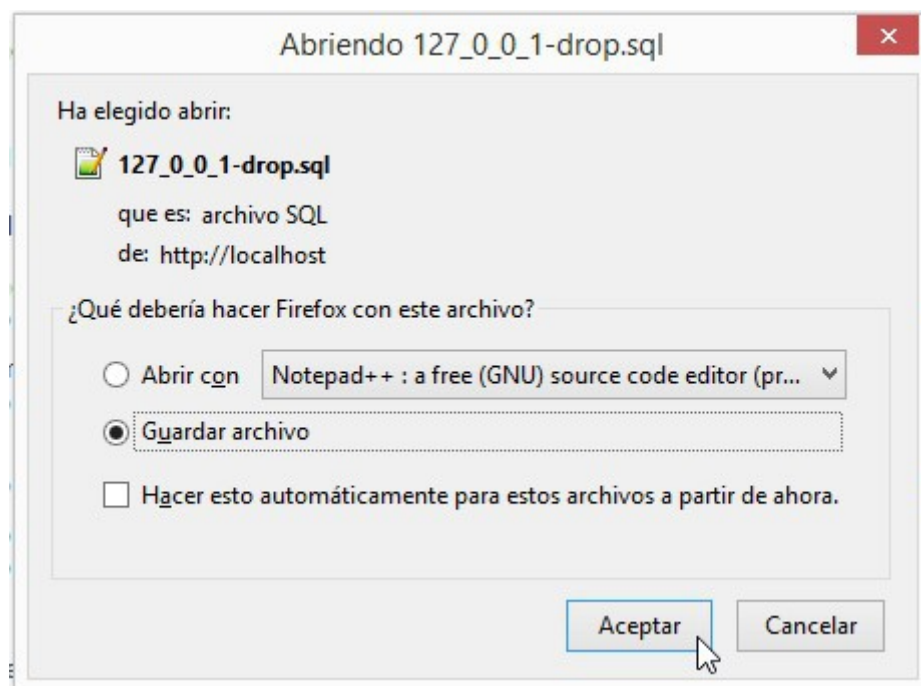
Opciones de creación de objetos







Agregar sentencias:

- ☐ Agregar sentencia DROP DATABASE
- ☒ Agregar sentencia DROP TABLE / VIEW / PROCEDURE / FUNCTION / EVENT / TRIGGER
- ☒ Agregar sentencia CREATE TABLE
- ☐ IF NOT EXISTS (menos eficiente porque los índices se generarán durante la creación de la tabla)
- ☒ Valor AUTO_INCREMENT
- ☒ Agregar sentencia CREATE VIEW
- ☒ Agregar sentencia CREATE PROCEDURE / FUNCTION / EVENT
- ☒ Agregar sentencia CREATE TRIGGER
- ☒ Encerrar nombres de tablas y campos con comillas invertidas (*protege los nombre de tablas y campos compuestos con caracteres especiales o palabras clave*)

- e) El resto de opciones podemos dejar las que vienen por defecto y pulsamos el botón **Continuar**. Pulsamos el botón Aceptar del cuadro de dialogo que se nos muestra para guardar en el disco duro del ordenador el archivo sql con la exportación de la base de datos (datos y estructura).

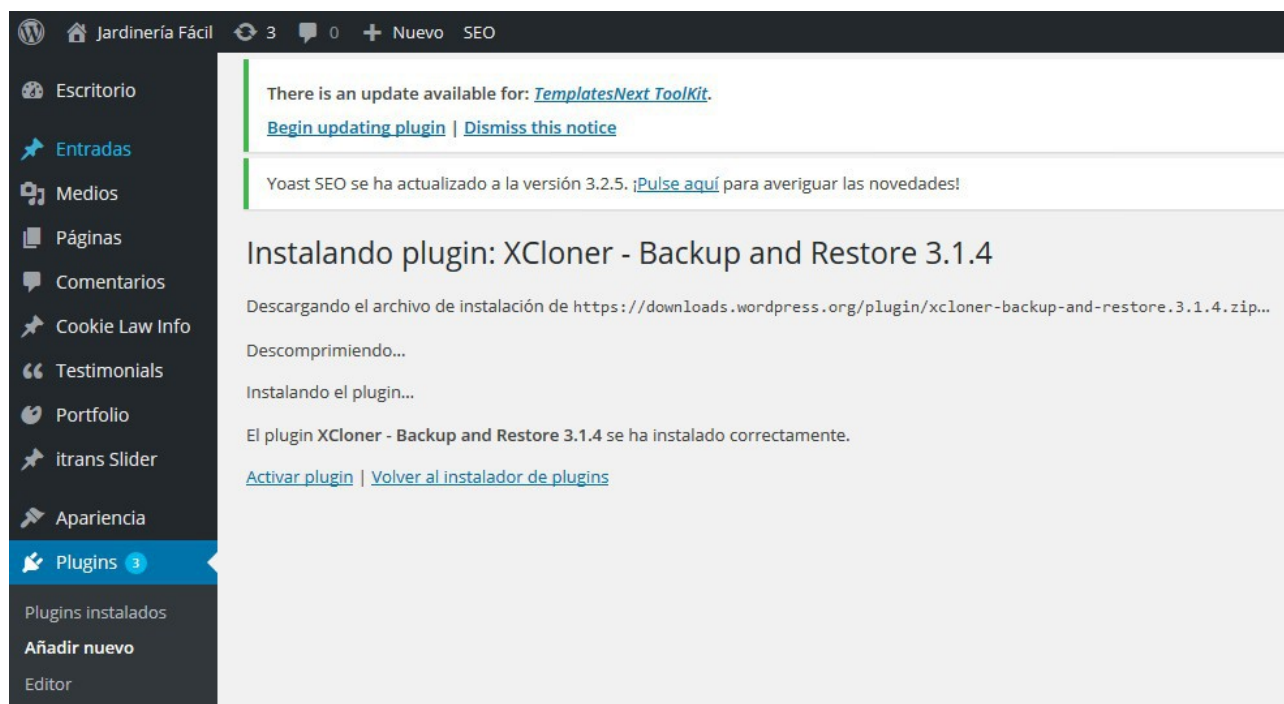
Continuar



Nombre	Fecha de modifica...
 127_0_0_1-drop.sql	30/08/2016 10:59
 database-sql.sql	29/08/2016 11:20
 gymforce-drop.sql	11/07/2016 11:18
 jardineriafacil.sql	24/08/2016 11:07
 jardineriafacil-localhost.sql	01/07/2016 10:06
 waterzone-drop.sql	22/06/2016 13:29

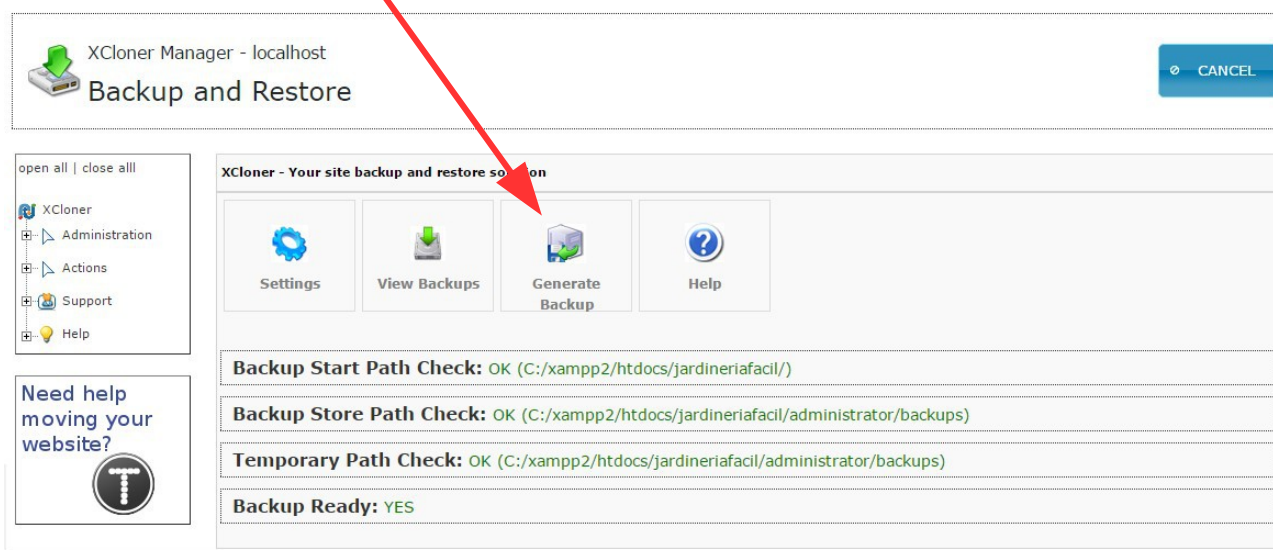
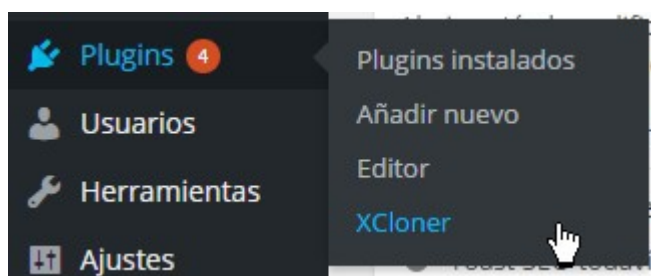
Backup con Plugin – XCloner

Sin embargo en WordPress disponemos de un estupendo plugin que nos simplifica la tarea de backup y de restauración. Instalamos dicho plugin Xcloner y lo activamos en el panel de administración de plugins de WordPress.



Una vez instalado y activado, tendremos una nueva opción en el menú Plugins para acceder a su pantalla de administración.

Para realizar el backup solo habrá que hacer clic en el botón **Generate Backup**

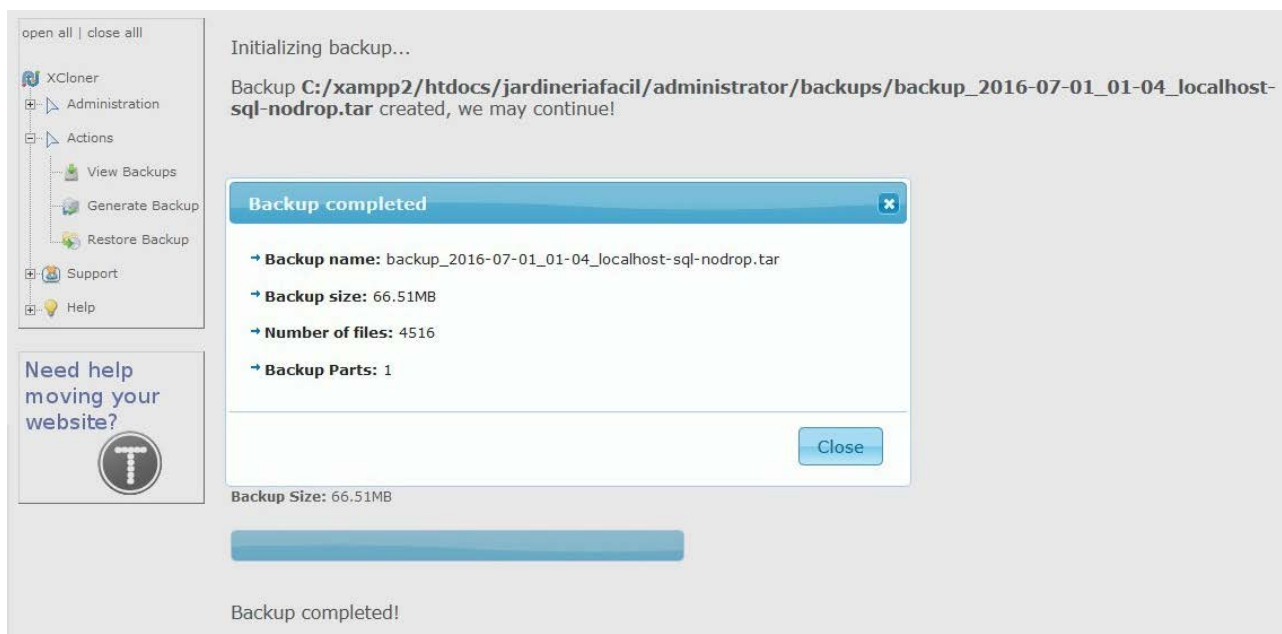


La siguiente pantalla es para realizar algún ajuste más como excluir tablas del backup o añadir alguna base de datos extras. En el proceso normal de backup de nuestra instalación de WordPress no haremos ningún cambio y solo pulsaremos el botón **Continue**.

A partir de ese momento el plugin empezará a crear el backup del WordPress en el que está instalado ...

... generando un archivo **.tar**, formato de fichero utilizado para crear un archivo en el que se incluye todos los ficheros y su estructura de directorios con opción de realizar compresión en su creación para reducir su tamaño. Dicho archivo tendrá como nombre **backup_fecha_hora-servidor-sql-nodrop.tar** que nos indica de qué servidor hemos realizado el backup, en qué fecha y a qué hora.

Cuando el proceso finalice veremos una ventana de información similar a la siguiente, en la que podemos comprobar el nombre del archivo de backup, su tamaño en MB y el número de archivos que contiene. Para salir pulsamos el botón **Close**.



En este paso ya tenemos una copia de seguridad de nuestro sitio web de WordPress, almacenada en un solo archivo .tar que se encontrará ubicado en **~/administrator/backups/**.

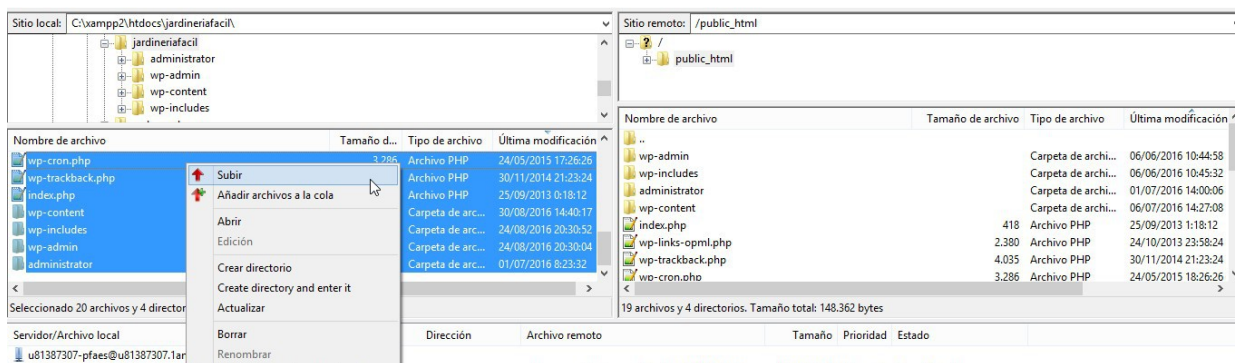
Restauración de la copia de seguridad

Un backup o copia de seguridad no tiene sentido si no realizamos una restauración a partir de ella en algún momento, por lo que siempre que hablamos de backup hay que conocer el proceso de restauración. En nuestro caso, el backup de WordPress, que como indicamos anteriormente es tan simple como realizar copia de los archivos en el que ésta alojada la instalación y de la base de datos en la que se almacenan los contenidos de la web. Es por lo que la restauración es tan simple como restauración de los archivos y la base de datos. Veamos como se hace tanto de forma manual o mediante la utilización de un plugin que automatiza y simplifica el proceso.

Restauración manual

Al igual que el backup la restauración manual la haremos en dos pasos:

1. Primero restaurando (copiando) los archivos de la instalación al servidor, ya sea por FTP si estamos llevando los archivos a un servidor remoto, o utilizando el administrador de ficheros si el servidor es nuestro servidor local.



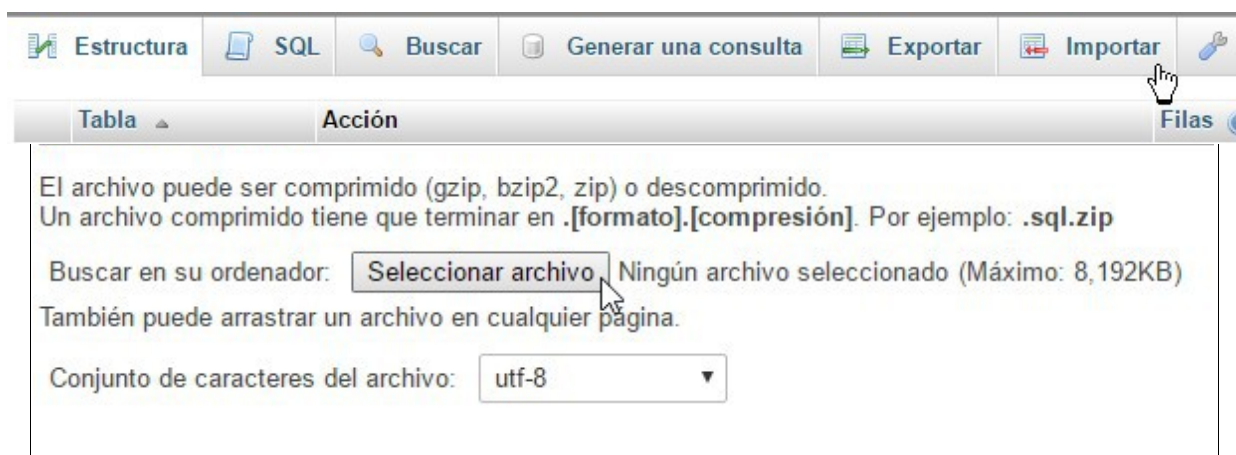
- Segundo importando en la base de datos del servidor la copia de la base de datos. Esta operación al igual que la de exportación de la base de datos, la vamos a realizar desde el administrador de phpmyadmin. Debemos tener en mente que en la mayoría de los alojamientos de páginas web que dan soporte de mysql no permiten crear bases de datos con sentencias sql, por lo que antes de importar la misma debemos realizar una pequeña modificación en el fichero que hemos exportado anteriormente. Esta modificación no es más que comentar la línea en la que se crea la base de datos. Lo que realizamos es eliminar todas las sentencias existentes desde el primer comentario del archivo hasta justo antes de la primera instrucción de CREATE TABLE, como vemos en la siguiente imagen resaltado en amarillo.

```

7  -- Versión del servidor: 10.1.10-MariaDB
8  -- Versión de PHP: 5.5.33
9
10 SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
11 SET time_zone = "+00:00";
12
13
14 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
15 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
16 /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
17 /*!40101 SET NAMES utf8mb4 */;
18
19 --
20 -- Base de datos: `wwwdeportesgaldar`
21 --
22 CREATE DATABASE IF NOT EXISTS `wwwdeportesgaldar` DEFAULT CHARACTER SET utf8 COLLATE utf8_ur
23 USE `wwwdeportesgaldar`;
24
25

```

Ahora, lo que haremos será conectarnos al administrador de MySQL del servidor remoto utilizando para ello phpmyadmin e **importar** la base de datos partiendo de este archivo.



Seleccionamos el archivo .sql que hemos editado y pulsamos el botón **Continuar**. Al finalizar la operación de importación obtendremos un mensaje como el que se muestra a continuación.

✓ La importación se ejecutó exitosamente, se ejecutaron 71 consultas. (127_0_0_1-drop.sql)

Estos son los dos pasos del proceso de restauración manual de la copia de seguridad. A partir de ese momento ya podremos acceder, si la copia se realizó en un momento en el que todo estaba funcionando correctamente, a la administración de nuestro WordPress y a la página web exactamente igual que estaba.

Restauración con Plugin – Xcloner

Para la restauración mediante el fichero de archivo creado por Xcloner, .tar, el paso inicial es subir al directorio en el que tengamos alojado o vayamos a alojar la web un directorio normalmente denominado **restore/** y que contendrá los siguientes archivos:

1. TAR.php
2. XCloner.php
3. index.html
4. backup...tar

Los dos primeros archivos los distribuye el propio plugin y los puedes encontrar en **~/wp-content/plugins/xcloner-backup-and-restore/restore/**

El archivo index.html, es un fichero básico de html que solo contiene el código `<html><head></head><body></body></html>` y cuya única función es evitar que el navegador muestre el contenido del directorio en el que está alojado.

El último archivo será el correspondiente fichero .tar resultado del backup realizado en algún momento con el plugin de Xcloner desde la administración de WordPress.

Estos archivos se copiarán a su ubicación usando el administrador de ficheros si es una instalación local o mediante FTP si es una instalación remota, ya sea en un ordenador en línea o en un ordenador de la red local que tenga activado el servidor de FTP, si no es así, en este último caso la opción más común es hacer una copia utilizando algún medio extraíble como un disco duro externo o un pendrive.

El siguiente paso es iniciar el script de restauración abriendo el navegador y escribiendo la dirección o URL que corresponda, en nuestro ejemplo:

<http://formacionit.esy.es/restore/XCloner.php>

Ten en cuenta que las mayúsculas y minúsculas son sensibles y hay que poner la dirección tal cual, respetando las mayúsculas. Al ejecutar esta URL lanzaremos el script que habíamos subido anteriormente al servidor e iniciaremos el proceso de restauración propiamente dicho.

XCloner Automatic Restore! - formacionit.esy.es

New Path & URL Configuration

1. Install directory: (server path where you want to restore your backup)

/home/u /public_html/restore/

2. Install Url: (the url correspondent of your Install directory:)

http:// formacionit.esy.es/restore

3. Choose backup archive to restore:

backup_2016-08-30_03-28_localhost-sql-nodrop.tar(69.39M) ▾

New Mysql Configuration

File Restore Options

Import mysql database

Incremental sql import

Mysql server: localhost

Mysql username:

Mysql password:

Mysql database:

Data encoding:

Default ▾

Start install

phpinfo()

*after hitting submit please wait for the package to get unarchived and transfered through ftp if it is the case!

Security Note: After restore **delete the XCloner.php script from your server**

XCloner Automatic Restore(3.1.0) - [Website Backup and Restore](#)

En esta ventana realizaremos el ajuste necesario siguiente:

- Install directory → eliminar restore/ del final de la url.
- Install Url → eliminar restore del final de la url.
- Mysql server → indicar el servidor de bases de datos en función de nuestra instalación.
- Mysql username → indicar el nombre de usuario de conexión para ese servidor.
- Mysql password → indicar la contraseña de usuario para ese servidor.
- Mysql database → indicar el nombre de la base de datos en el que se volcarán los datos del backup.

Quedando en nuestro caso como se muestra en la imagen siguiente:

XCloner Automatic Restore! - formacionit.esy.es

New Path & Url Configuration

1. Install directory: (server path where you want to restore your backup)

/home/u /public_html/

2.Install Url: (the url correspondent of your Install directory:)

http:// formacionit.esy.es

3. Choose backup archive to restore:

backup_2016-08-30_03-28_localhost-sql-nodrop.tar(69.39M)

New Mysql Configuration

File Restore Options

Import mysql database

Incremental sql import

Mysql server:

mysql.hostinger.es

Mysql username:

u _root

Mysql password:

123456

Mysql database:

u _egard

Data encoding:

Default

Start install

phpinfo()

*after hitting submit please wait for the package to get unarchived and transfered through ftp if it is the case!

Security Note: After restore **delete the XCloner.php** script from your server

XCloner Automatic Restore(3.1.0) - [Website Backup and Restore](#)

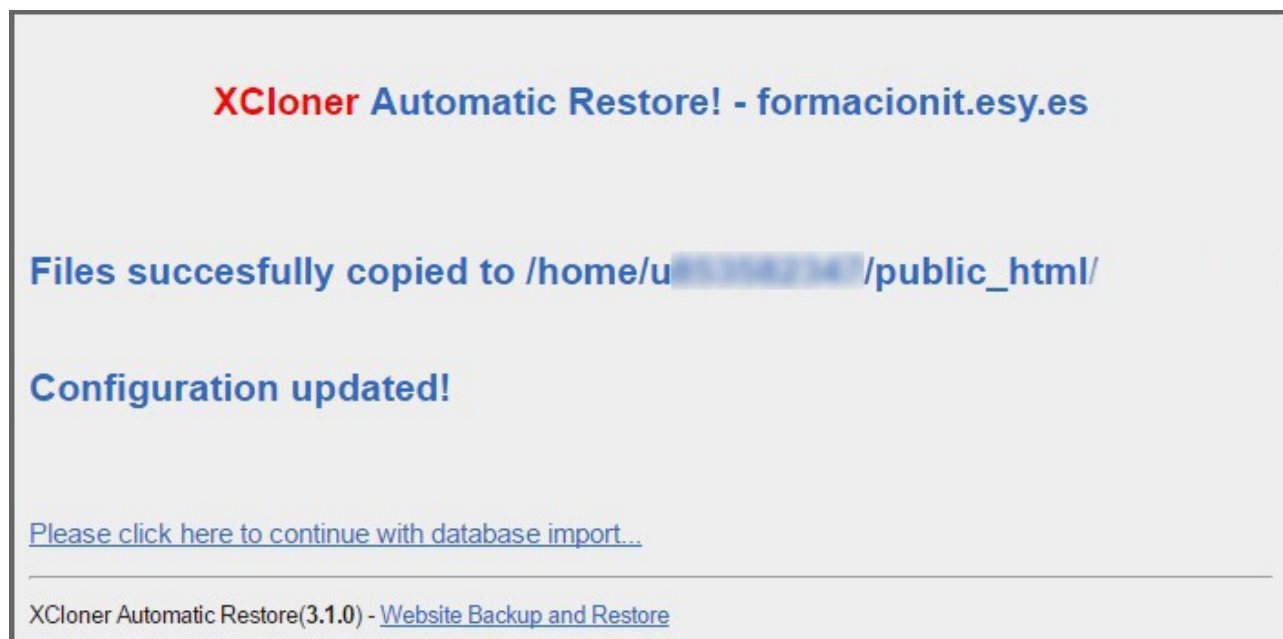
Siguiente paso, hacer clic en el botón **Start install**. Esta acción iniciará el proceso de descompresión del archivo **.tar** indicado y la restauración del backup.

XCloner Automatic Restore! - formacionit.esy.es

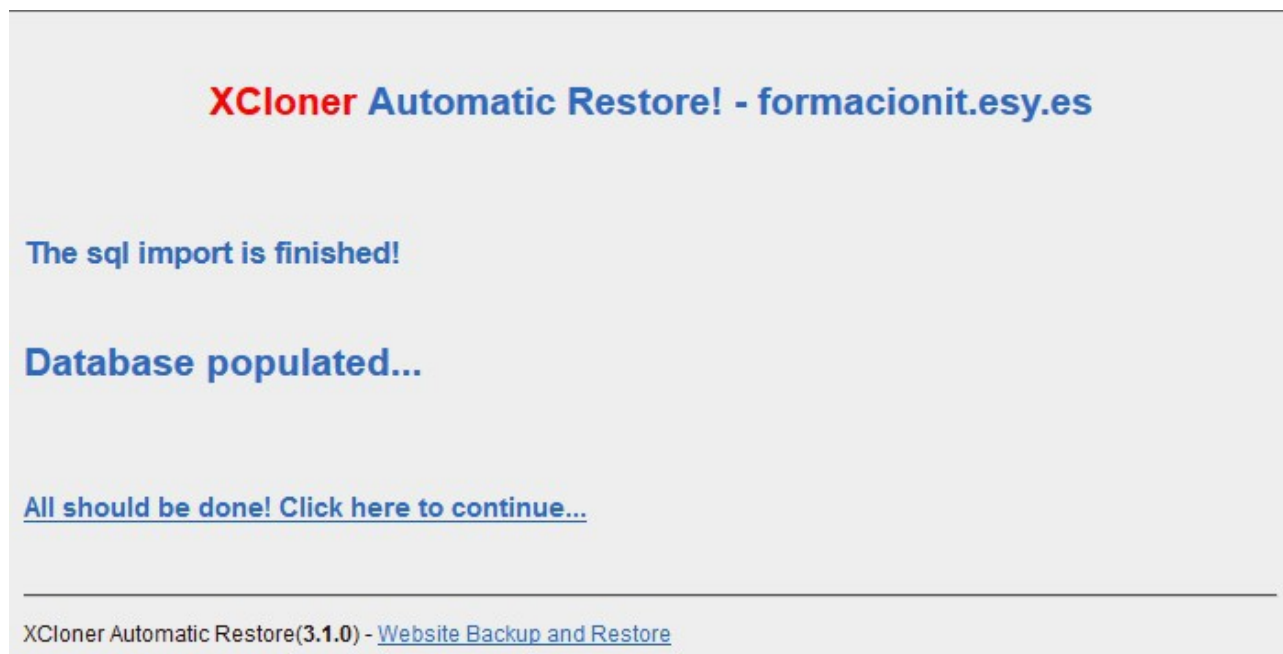
Processing files - 12.93%

Al finalizar, la primera parte del proceso (descompresión de los archivos) aparecerá la

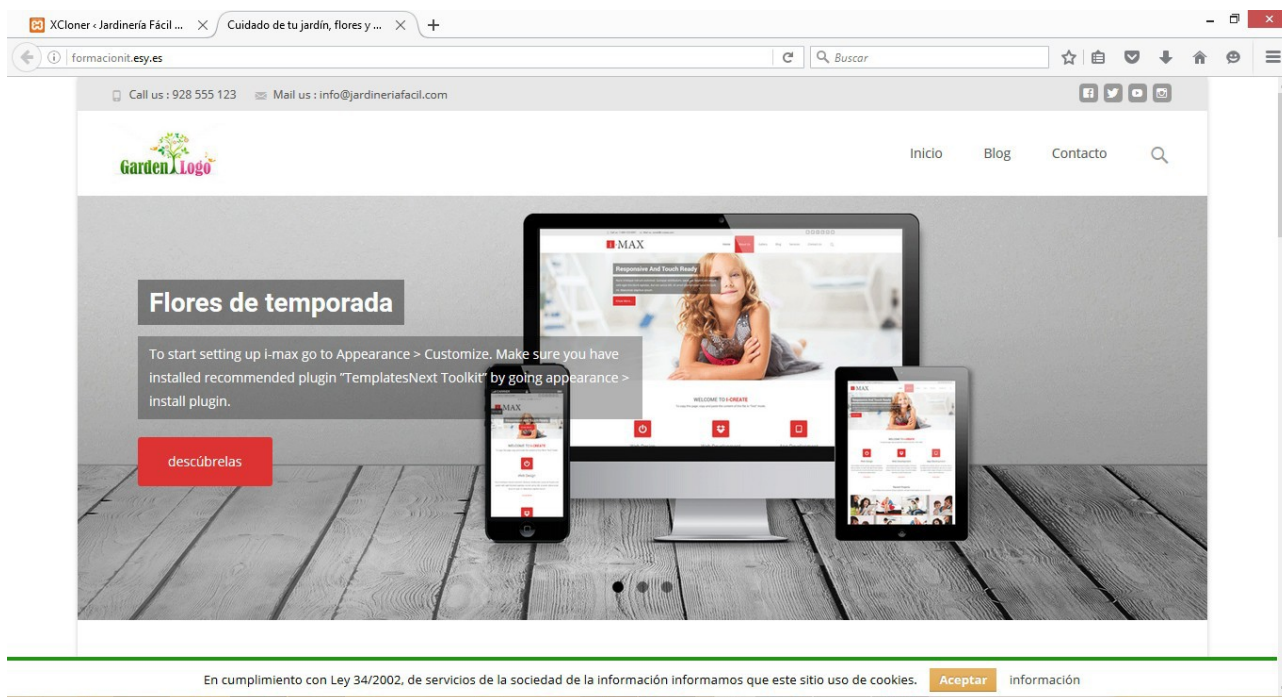
siguiente pantalla:



En la que deberemos hacer clic en el enlace ***“Please click here to continue with database import...”*** para continuar con el siguiente proceso que será el de volcar los datos de la base de datos. El script informará de la finalización del volcado de datos a la base de datos con una pantalla similar a la siguiente:



Al hacer clic en el enlace de la pantalla ***“All should be done! Click here to continue...”*** nos llevará a nuestra página web recién restaurada. 😊



Ajustes de WordPress después de una restauración

Hasta aquí hemos explicado paso a paso las tareas de realización de copia de seguridad (backup) y restauración de la misma.

Primera acción post-restauración

La primera acción que debemos realizar después de restaurar nuestra copia de seguridad es **eliminar** de nuestro hosting el directorio **/restore**. Para que, por una parte no dejar una brecha de seguridad dejando el script Xcloner.php accesible, y por otro lado eliminando el fichero .tar que acabamos de restaurar, puesto que si no lo hacemos y más adelante realizamos una copia de seguridad ese directorio /restore se incluirá en el fichero .tar aumentando considerablemente el tamaño del nuevo archivo .tar.

Ajustes de configuración por cambio de ubicación del servidor

En este proceso hemos asumido que restauramos la copia de seguridad en la misma máquina en la que hicimos el backup, sin embargo, en otras muchas ocasiones la restauración se realizará en otra máquina, como ya indicamos al inicio de este humilde tutorial. En estos casos en los que la restauración del WordPress se va a realizar en otra ubicación hemos de realizar tres sencillas tareas de ajuste para que la web funcione correctamente.

La situación más común es el traslado de nuestro sitio web desde nuestro servidor de desarrollo local (localhost) a un servidor en internet (hosting). Este es el caso que vamos a utilizar para explicar los ajustes a realizar post-restauración, pero que son exactamente iguales ya sea la restauración en otra máquina local de nuestra red, en otro servidor de hosting distinto, etc Empezamos...

Editar wp-config.php

El fichero wp-config.php contiene la configuración básica del WordPress como los parámetros de conexión a la base de datos y variables de ejecución. Cuando realizamos backup y restauraciones con Xcloner, el script añade varias líneas en las que establece las variables **WP_SITEURL**, **WP_HOME** y **RELOCATE**, sin embargo no elimina las ya existentes, por lo que editaremos el archivo del servidor y lo volveremos a sustituir por FTP eliminando dichas líneas. En la imagen que se muestra a continuación:



```
34
35  /** MySQL hostname */
36  define('DB_HOST', 'mysql.hostinger.es');#localhost');
37
38  define('WP_SITEURL','http://formacionit.esy.es');
39  define('WP_HOME','http://formacionit.esy.es');
40  define('RELOCATE',true);
41
42
43  define('WP_SITEURL','http://localhost/jardineriafacil');
44  define('WP_HOME','http://localhost/jardineriafacil');
45  define('RELOCATE',true);
46
47
48
49  /** Database Charset to use in creating database tables. */
50  define('DB_CHARSET', 'utf8mb4');
```

Como puedes observar, esas mismas definiciones existen para nuestro equipo de desarrollo, localhost, y Xcloner ha añadido las nuevas con la ubicación del hosting de destino, formacionit.esy.es. Eliminamos las líneas indicadas con el recuadro amarillo, guardamos los cambios del archivo y lo volvemos a subir a nuestro hosting por FTP sobrescribiendo el fichero en el servidor.

Editar la tabla _options

Estos mismos valores hay que cambiarlos también en la tabla **_options**. Para hacerlo nos conectamos a través del panel de administración del hosting a la herramienta phpmyadmin a esa base de datos, como podemos observar en la imagen adjunta:

Examinar Estructura SQL Buscar Insertar Exportar Importar Operaciones Seguir

Mostrando registros 0 - 29 (191 total, La consulta tardó 0.0004 seg)

```
SELECT *
FROM 'jf_options'
LIMIT 0, 30
```

Perfilando [En línea] [Editar]

Mostrar : Fila de inicio: 30 Número de filas: 30 Cabeceras cada 100 filas

Ordenar según la clave: Ninguna

+ Opciones

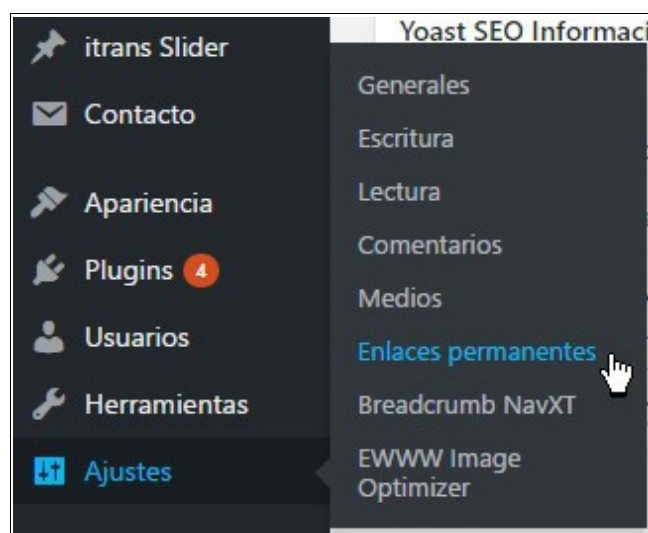
	option_id	option_name	option_value	autoload
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	1	siteurl	http://localhost/jardineriafacil	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	2	home	http://localhost/jardineriafacil	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	3	blogname	Jardinería Fácil	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	4	blogdescription	Jardinería, paisajismo, herramientas y consejos	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	5	users_can_register	0	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	6	admin_email	octavioh@ironika.es	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	7	start_of_week	1	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	8	use_balanceTags	0	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	9	use_smilies	1	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	10	require_name_email	1	yes
<input type="checkbox"/> Editar <input type="checkbox"/> Copiar <input type="checkbox"/> Borrar	11	comments_notify	1	yes

Editamos ambos registros y sustituimos el valor actual de <http://localhost/jardineriafacil> por el nuevo valor <http://formacionit.esy.es>

Restaurar Enlaces Permanentes

La última acción de ajuste ya la vamos realizar en el back-end, panel de administración de nuestro WordPress, de nuestro hosting en internet. Para ellos abrimos el navegador y escribimos la dirección URL del escritorio de administración de WordPress. Siguiendo nuestro ejemplo, sería <http://formacionit.esy.es/wp-admin>

Utilizando una credencial de administración, nombre de usuario y contraseña, válidos accedemos al panel de administración y hacemos clic en **Ajustes** → **Enlaces Permanentes**. Una vez en esta pantalla se comprueba que está activa la opción **Enlaces Permanentes** y se pulsa el botón **Guardar**.



☒ **Nombre de la entrada**

☐ **Estructura personalizada**

Opcional

Si quieres, puedes introducir aquí estructuras personalizadas para las URLs de etiqueta o categoría algo como . Si dejas esto en blanco se usará la estructura por defecto.

Categoría base

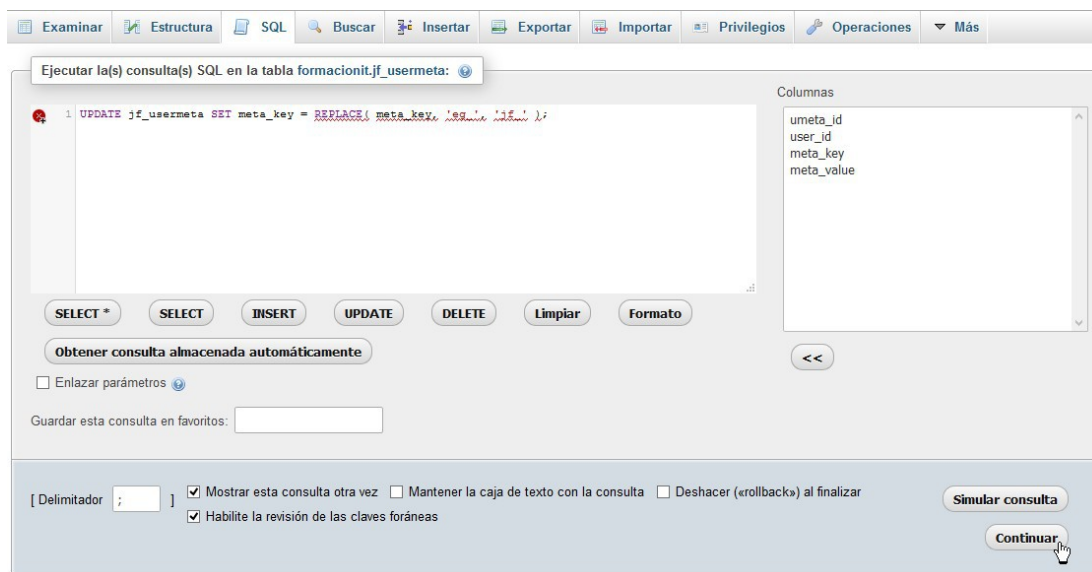
Etiqueta base

Cambio de prefijo

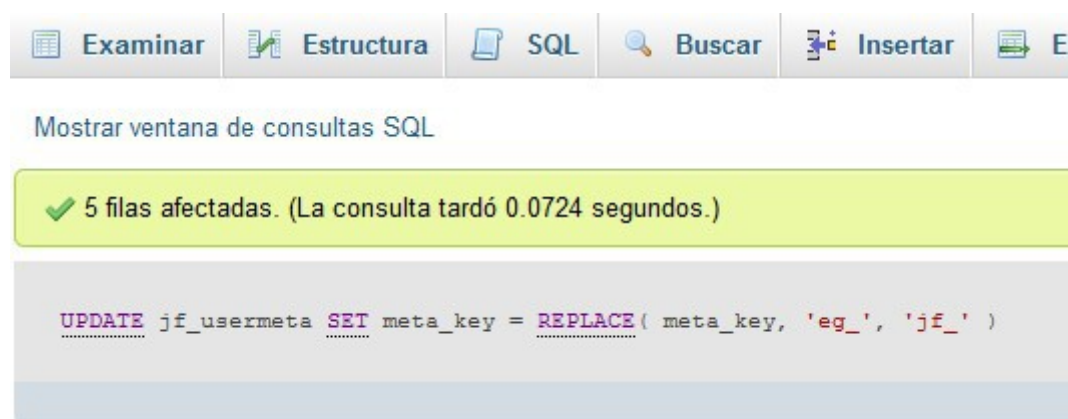
Cuando movemos una web de una localización a otra, a veces es necesario modificar manualmente valores en la base de datos para hacer que la nueva URL sea reconocida adecuadamente. Esto es lo que hemos hecho en los tres pasos anteriores post-restauración, sin embargo existe una situación que también se te puede dar y que requiere que realices manualmente el siguiente ajuste. La situación que queremos corregir es cuando el prefijo de las tablas de datos de wordpress cambian de un servidor a otro. Es una situación poco común pero si te ves en ella la forma de resolverlo es la siguiente:

1. Abre el administrador de phpmyadmin del nuevo hosting y selecciona la base de datos que contiene los datos de WordPress si no está ya seleccionada.
2. Selecciona la tabla ***_usermeta**, dónde * es el nuevo prefijo de tabla de esta instalación y pulsa el botón SQL del panel de administración de phpmyadmin.
3. Escribe la siguiente instrucción SQL, adaptándola a tu situación en ese momento, para este ejemplo vamos a cambiar el prefijo antiguo de eg_ al nuevo prefijo de jf_:

UPDATE eg_usermeta SET meta_key = REPLACE(meta_key, 'eg_', 'jf_');



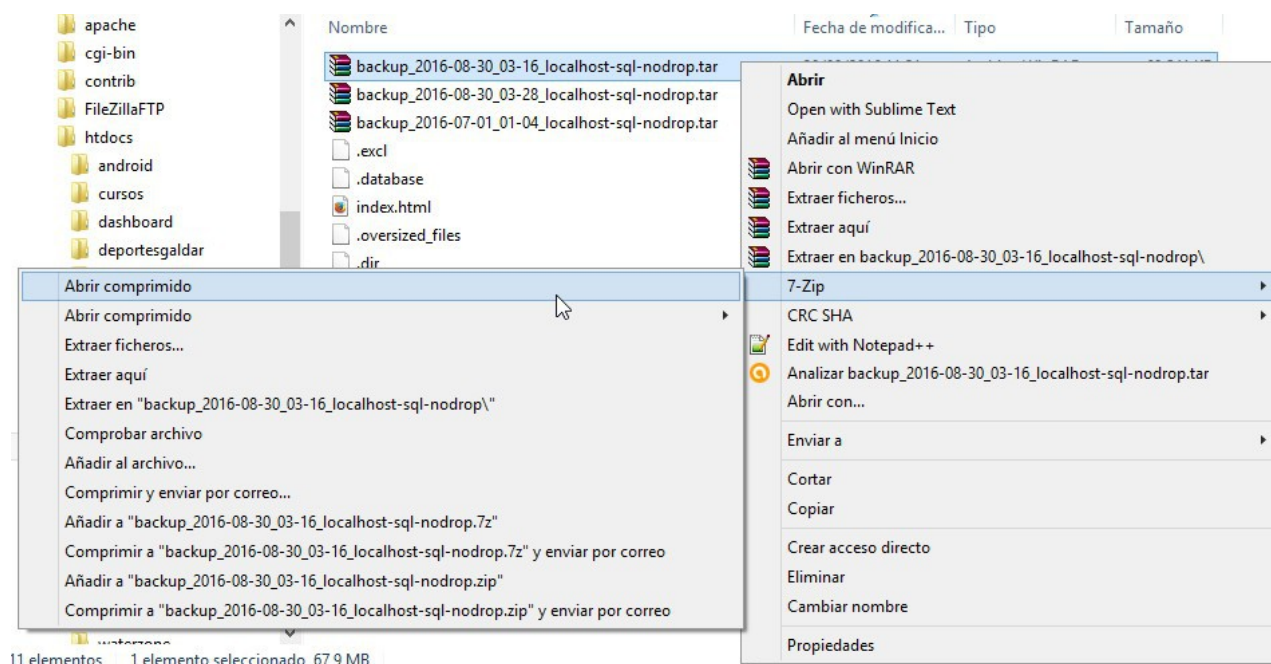
La ejecución de dicha instrucción debe dar como resultado la sustitución de dichos prefijos y una ventana de confirmación de realizado como vemos en la imagen siguiente:



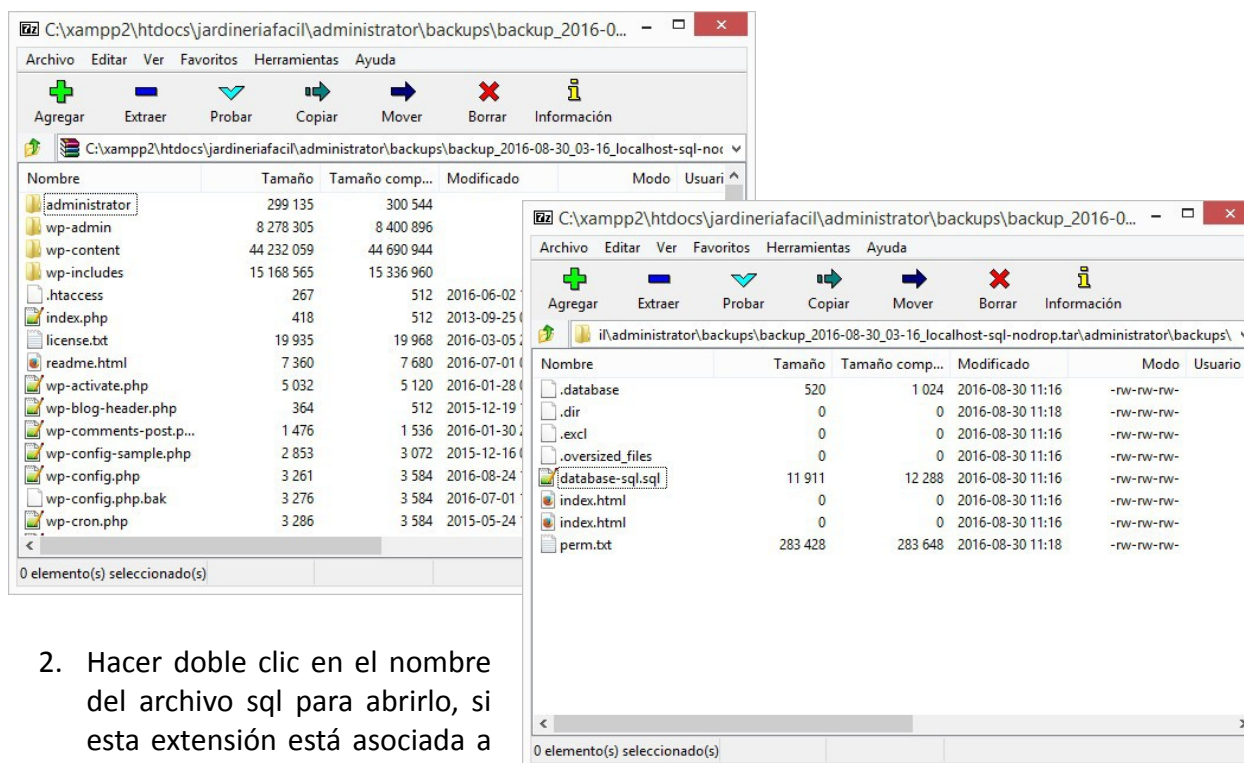
Precauciones con el plugin Xcloner y localhost

Hemos descubierto que el plugin Xcloner no realiza el volcado de los contenidos de la base de datos en el proceso de backup cuando se ejecuta en local (localhost), creando simplemente un volcado de la estructura de la base de datos. Esto lo puedes comprobar de la siguiente manera:

1. Abre, preferiblemente con 7zip ya que es una herramienta de software libre y te permitirá en un paso posterior corregir este problema, aunque también puedes hacerlo con winzip o winrar, pero ninguno de ellos, a menos que dispongas de la versión registrada podrás editar el archivo, el archivo .tar que ha generado el Xcloner al realizar el backup y que se encuentra en **~/administrator/backups**

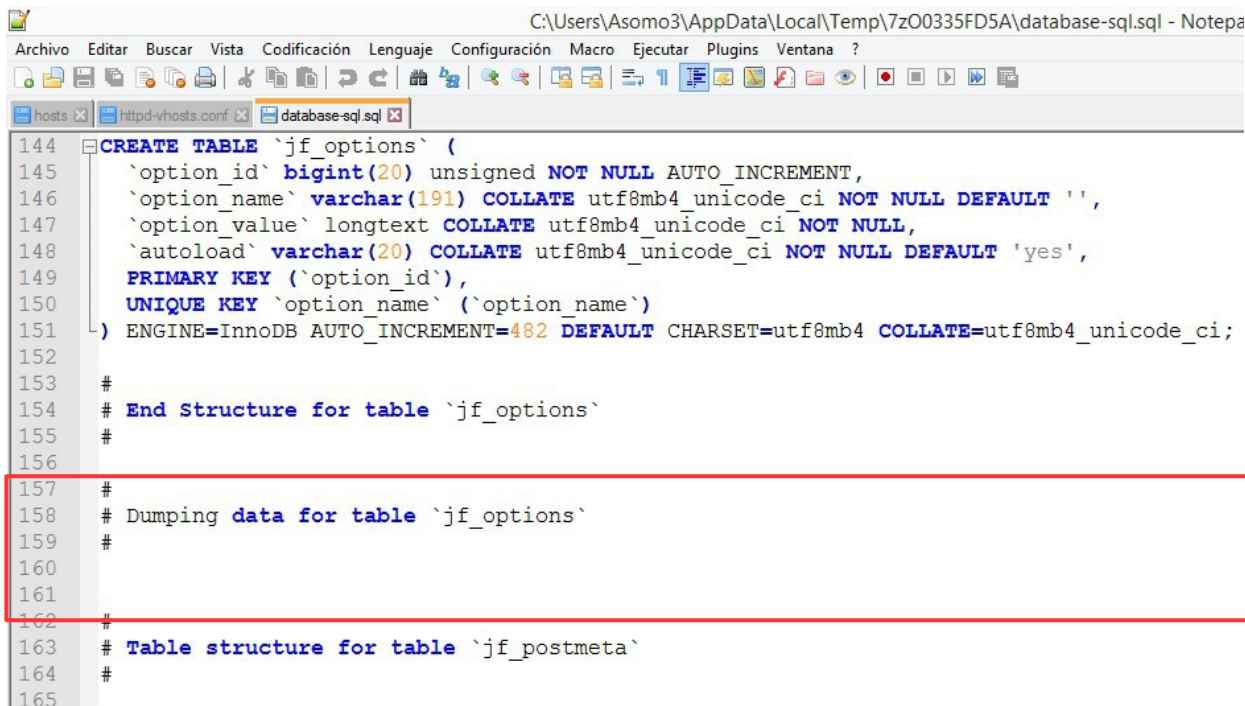


En la ventana del 7zip, navega haciendo doble clic en las carpetas, hasta llegar al directorio que contiene el archivo sql con la exportación de la base de datos y cuyo nombre será **database-sql.sql**, como se observa en las imágenes siguientes:



- Hacer doble clic en el nombre del archivo sql para abrirlo, si esta extensión está asociada a algún programa en cuestión, como Note++ se abrirá en él, sino el sistema pide que se indique con qué aplicación abrir este tipo de archivos. Busca la instrucción sql de por ejemplo la **tabla _options** y

comprueba si existen instrucciones **INSERT** a partir del comentario **# Dumping data for table `*_options`**. Si no existe ninguna instrucción **INSERT** en esta tabla es que el plugin Xcloner está fallando y pasaremos a resolver manualmente esta situación generando un nuevo backup de la base de datos y sustituyendo este en el archivo .tar. Lo vemos a continuación.



```
144 CREATE TABLE `jf_options` (  
145   `option_id` bigint(20) unsigned NOT NULL AUTO_INCREMENT,  
146   `option_name` varchar(191) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT '',  
147   `option_value` longtext COLLATE utf8mb4_unicode_ci NOT NULL,  
148   `autoload` varchar(20) COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT 'yes',  
149   PRIMARY KEY (`option_id`),  
150   UNIQUE KEY `option_name` (`option_name`)  
151 ) ENGINE=InnoDB AUTO_INCREMENT=482 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;  
152  
153 #  
154 # End Structure for table `jf_options`  
155 #  
156  
157 #  
158 # Dumping data for table `jf_options`  
159 #  
160  
161  
162 #  
163 # Table structure for table `jf_postmeta`  
164 #  
165
```

Resumiendo lo que se hace es:

- Exportar con phpmyadmin la base de datos su estructura y datos.
- Eliminar del archivo resultante líneas de comentarios que no son necesarios para la restauración con Xcloner.
- Renombrar el archivo generado a database-sql.sql
- Sustituir el archivo database-sql.sql del archivo .tar de backup de Xcloner por este usando la aplicación 7zip.

Vemos a continuación esta tarea mediante capturas de pantalla.

Exportar la estructura y datos de la base de datos con phpmyadmin

Abre el administrador de phpmyadmin de tu máquina local, localhost, selecciona la base de datos a exportar y pulsa el botón Exportar de la barra de herramientas:

Exportando tablas de la base de datos "easygarden"

Exportar plantillas:

Nueva plantilla:

Nombre de plantilla

Crear

Plantillas existentes:

Plantilla: -- Seleccionar plantilla --

Actualizar

Borrar

Método de exportación:

- ☐ Rápido - mostrar sólo el mínimo de opciones de configuración
- ☒ Personalizado - mostrar todas las opciones de configuración posibles

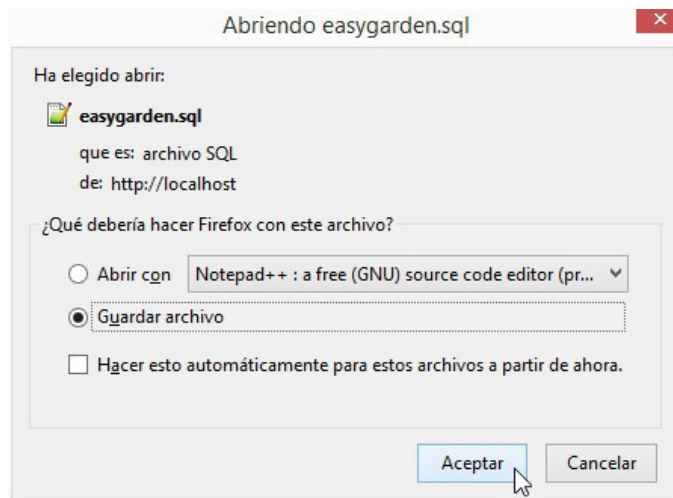
Seleccionar el método de exportación Personalizado y selecciona la opción *Agregar sentencia DROP TABLE* en el apartado de Opciones de creación de objetos.

Opciones de creación de objetos

Agregar sentencias:

- ☐ Agregar sentencia CREATE DATABASE / USE
- ☒ Agregar sentencia DROP TABLE / VIEW / PROCEDURE / FUNCTION / EVENT / TRIGGER
- ☒ Agregar sentencia CREATE TABLE
- ☐ IF NOT EXISTS (menos eficiente porque los índices se generarán durante la creación de la tabla)
- ☒ Valor AUTO_INCREMENT
- ☒ Agregar sentencia CREATE VIEW
- ☒ Agregar sentencia CREATE PROCEDURE / FUNCTION / EVENT
- ☒ Agregar sentencia CREATE TRIGGER
- ☒ Encerrar nombres de tablas y campos con comillas invertidas (protege los nombre de tablas y campos compuestos con caracteres especiales o palabras clave)

Hacer clic en el botón **Continuar** del final de la página y en el botón **Aceptar** del cuadro de diálogo que se mostrará a continuación seleccionando previamente la opción *Guardar archivo* para descargar el archivo sql generado a la carpeta de descargas que tengamos configurada en nuestro equipo.



Editar archivo sql generado

Abrir el archivo sql con Notepad++ o cualquier otro editor de archivos y eliminar las líneas resaltadas en las siguientes imágenes del principio del archivo y del final.

```
1  -- phpMyAdmin SQL Dump
2  -- version 4.5.1
3  -- http://www.phpmyadmin.net
4
5  -- Servidor: 127.0.0.1
6  -- Tiempo de generación: 06-09-2016 a las 10:33:59
7  -- Versión del servidor: 10.1.10-MariaDB
8  -- Versión de PHP: 5.5.33
9
10 SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
11 SET time_zone = "+00:00";
12
13
14 /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
15 /*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
16 /*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
17 /*!40101 SET NAMES utf8mb4 */;
18
19
20 -- Base de datos: `easygarden`
21
22
23
24
```

Guardar el archivo para que los cambios sean permanentes.

```
1098 ALTER TABLE `jf_users`
1099 MODIFY `ID` bigint(20) UNSIGNED NOT NULL AUTO INCREMENT, AUTO INCREMENT=2;
1100
1101 /*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
1102 /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
1103 /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

Renombrar el archivo sql generado

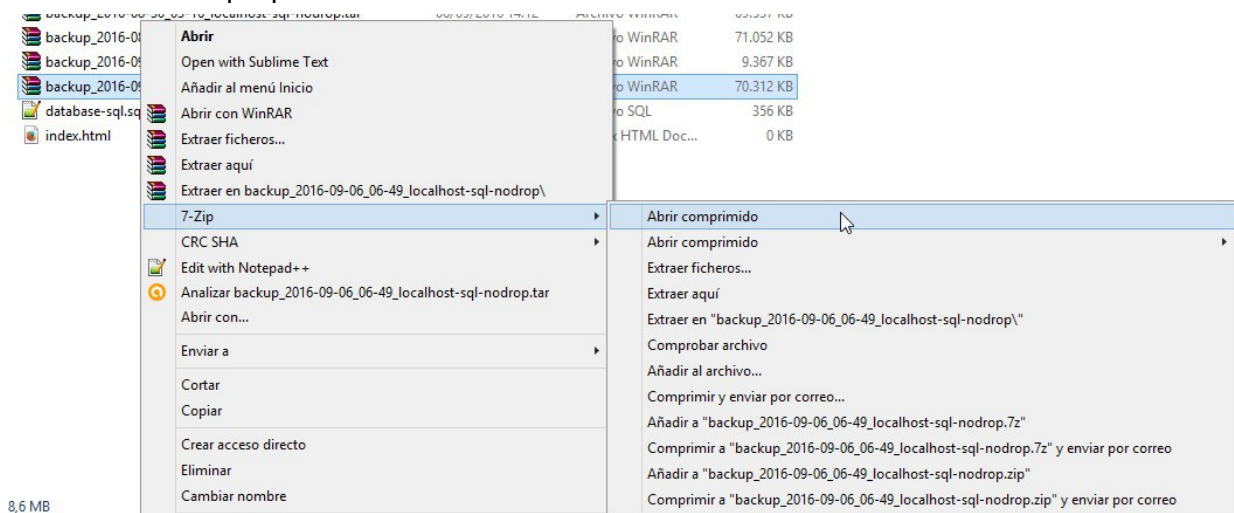
easygarden.sql	06/09/2016 9:34	Archivo SQL	356 KB
index.html	06/09/2016 14:49	Firefox HTML Doc...	0 KB

Cambiar el nombre de este archivo por el de **database-sql.sql**.

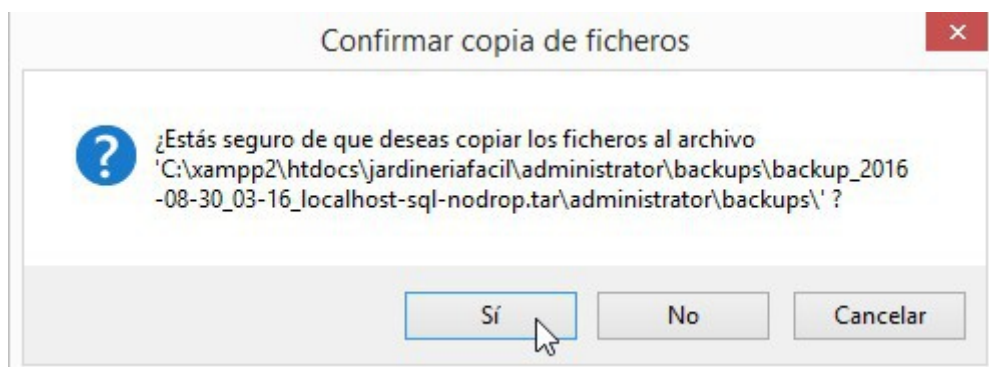
database-sql.sql	06/09/2016 9:34	Archivo SQL	356 KB
index.html	06/09/2016 14:49	Firefox HTML Doc...	0 KB

Sustituir el archivo erróneo por el nuevo en el .tar

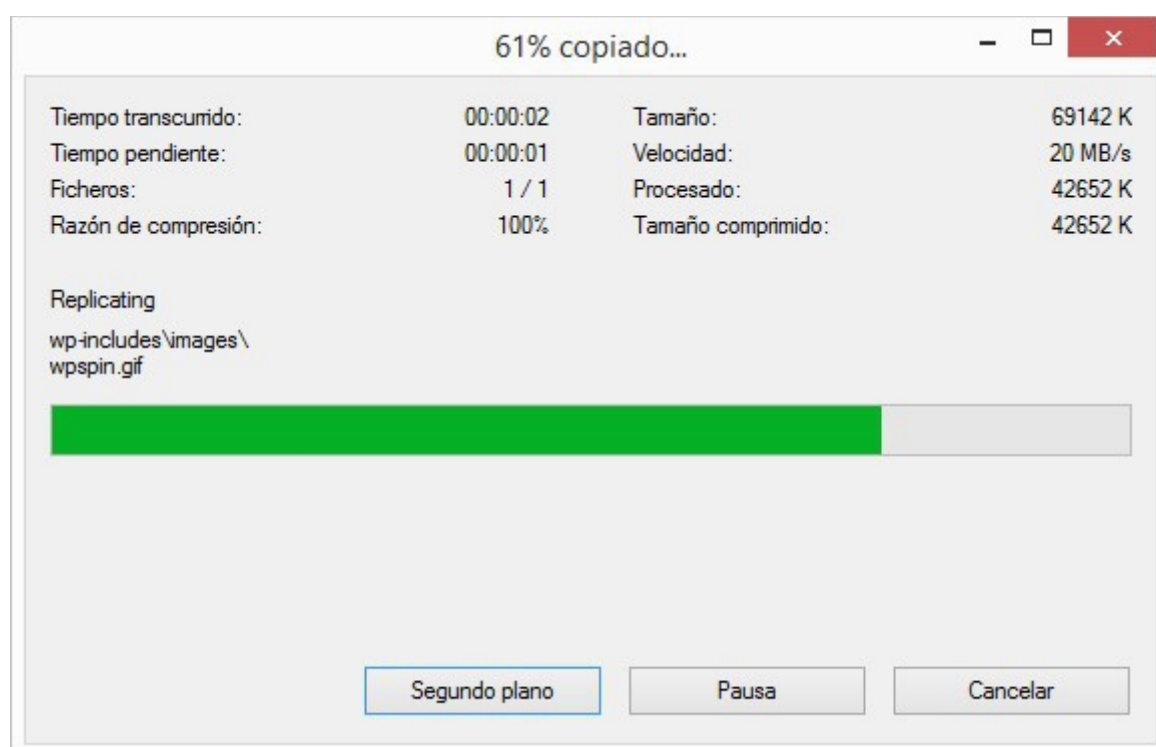
Este último paso lo haremos abriendo el .tar con 7zip y arrastrando y soltando el archivo nuevo database-sql.sql a la ventana.



Hacer clic en el botón **Si** para confirmar la copia de ficheros



El proceso habrá finalizado cuando la barra de progreso llegue al 100%...



Ahora ya disponemos de un archivo .tar correcto para realizar la restauración del backup de Xcloner ya sea en un equipo local o un equipo remoto en un hosting de internet.

Nota: Hasta el momento el plugin Xcloner no muestra este fallo cuando lo hemos ejecutado en un hosting en internet, pero nunca está de más realizar la comprobación del volcado de la base de datos al realizar nuestro primer backup con Xcloner, además de comprobar la integridad del fichero .tar para que no existan sorpresas al restaurar la copia en el peor momento.