EXT: KB Packman

Extension Key: **kb_packman**

The content of this document is related to TYPO3

- a GNU/GPL CMS/Framework available from www.typo3.com

Table of Contents

Introduction

What does it do?

This extension enables a backend user to unpack (decompress) compressed files in the fileadmin folder (via the File / Filelist Module). Currently “zip”, “tar.gz” (“tgz”) and “tar.bz2” (“tbz2”) files are supported. If there is requirement for it also “rar” file support will be added. Additionally to the precursor of this extension “kb_unpack” this extension also allows to create/compress a directory or a single file into a zip archive.

Users manual

After installing the extension via the Extension Manager it will work without any changes required.

Uploading the compressed file

Go to the “Filelist” Module (In the left Backend menu in the section “Files”). If you are a normal user your administrator could have disabled this menu entry for you. Ask him to enable this feature for you because you would like to upload files to the fileadmin.

If you are in the Filelist module left-click on the icon of a directory. A context menu will pop up. In this menu select “Upload Files”. Now you get presented a page where you can upload up to 10 files at a time. Upload your compressed files by selecting them from your harddrive via the “Browse” Button. If you selected all your files click “Upload files”. You can also mark the checkbox “Overwrite existing files” so already existing files with the same name get overwritten.

Unpacking the compressed file

Now if you have your file uploaded you will see it in the filelist. If you didn't check “Overwrite existing files” and a file with this name already existed your new uploaded file will get appended a number “_2”.

Now click on the Icon of the compressed file. A context menu will open which shows “Unpack” and “Unpack (OVERWRITE)” as the last two entries.

If you click “Unpack” the file will get decompressed but all already existing file will be kept. If you click “Unpack (OVERWRITE)” existing files will get overwritten.

When you clicked one of the unpack menu items you will get to a page where it is shown which files were actually extracted. It can happen that you get to an page which displays an error message stating that you are not allowed to extract files with .php files in it.

The reason why the extraction of “.php” files is inhibited for non-admins is security. A user could upload a malicious php file which could directly modify the database for example. So this is disabled for security reasons.

Creating an archive

When you open the click-menu (context-menu) of an file or directory you have a new option “Create zip”. Selecting this option will create a zip archive of the selected file or directory in the location specified by the Administrator of the site when installing this extension.

Administration

The only administrative action a administrator must take to allow a user to unpack files is to set the “Files: Unzip” Flag for this user. To do this log in as admin. Go to the List module and open the site-root where the backend users are stored. Click the Edit icon of the user for which you want to allow unpacking. Go to the bottom of the user config. There you see a section called “Fileoperation permissions”. In this section there is an option/checkbox “Files: Unzip”. Check this checkbox to allow unpacking to the user. Admins have the privilege of unpacking automatically.

When installing the extension there are a few options to configure in the Extension Manager:

Compress directory target-folder
Compress file target-folder

Those options specify where newly created archives get stored. When the “directory” option is set to “./” the archive of an directory will get placed into the directory itself. When the “file” option is set to “./” the created archive will get placed into the same directory where the file which got compressed resided.

The behaviour for directories - to put the archive into the directory being compressed itself - is a little bit unusual. If you want to place the archive of an directory into the the parent of the compressed directory just set the value to “../”

You could also set those values to a path relative to the TYPO3 site root. i.e: “fileadmin/myzips/” - this will result in all created archives being placed in this directory.

The option

DAM support

Will result in each compressed/decompressed file to being indexed by DAM automatically. You can also set default-values for the options available for DAM. (See the file README.txt in the extension directory for additional instructions. If you create a nice manual-paragraph from this information drop me a mail with your text - I will add it to this manual :)

Configuration

Normally it is not required to configure something for KB Packman. But if your “unzip” binary is in some abnormal location and can't get reached via the “PATH” variables then it is required to configure a little bit. Also if you “unzip” is somehow different from the standard UNIX unzip there will be some configuration required.

Setting up the unzip path

If your “unzip” binary is in some abnormal location you will need to configure where it is. Go to the install tool of Typo3. Go to “All Configuration”. Search for “unzip”. You will find an entry like:

```
[BE][unzip_path] = /usr/bin/unzip
```

Set this value to the correct path to your unzip binary.

Setting up for different unzips

This configuration is especially for “zip” files. Tar has a standard interface which is common to most UNIXes. So just for unzip special configuration parameters have to get set. This parameters get set in the “ext_localconf.php” file of the extension (You find this file in the extension directory). After chaning values in this file you will have to clear temp_cached(The red “clear-cache” button), else the changes will not get applied.

NOTICE: This step is not necessary in most cases. So just read on if you experience strange behaviour (Missing files, etc.) during usage of kb_packman.

It is not really necessary to configure this and the extension will also work without it but the recognition of “.php” files and the listing of the files which got extracted will not be valid without checking that your configuration is OK.

To test if your unzip is compliant you will need shell access to the server where your Typo3 is installed. (Of course for this task it is not really required to have shell access but i makes life easier. Without shell access you will have to copy over “zip” and “unzip” to your own server and look there how they act. But this could lead to problems with missing libraries.)

Create sample zip file

The first step is to create a sample zip test file which you will use for testing. Use the “zip” command to create a simple zip file containing 3-4 files.

The list configuration

Type the following command on the shell in the directory where you created your simple zip file:

```
unzip -t my_zipfile.zip
```

If the output from this command looks EXTACTLY like the one below you are safe:

```
Archive:  test.zip
  Length     Date   Time    Name
 --------    ----   ----    ----
        0  01-04-05 13:01   1.html
        0  01-04-05 13:01   2.html
        0  01-04-05 13:01   3.html
        0  01-04-05 13:01   4.html
 --------                   -------
        0                   4 files
```

There are 3 lines of text before the real filelisting. These lines are:

```
Archive:  test.zip
  Length     Date   Time    Name
 --------    ----   ----    ----
```

If you have more or less than those 3 lines before the start of the filelisting then you will need to configure the following variable.

```
$TYPO3_CONF_VARS["BE"]["unzip"]["list"]["pre_lines"] = XX;
```

If you have more or less than 2 lines after the file listing you will need to configure the following variable:

```
$TYPO3_CONF_VARS["BE"]["unzip"]["list"]["post_lines"] = XX;
```

Then you will have to take a look at the listing itself. This is an example line of the file listing:

```
        0  01-04-05 13:01   3.html
```

The listing gets separated by spaces. The 4th element is the name of the file. So you will need to do the following setup:

```
$TYPO3_CONF_VARS["BE"]["unzip"]["list"]["split_char"] = " ";
$TYPO3_CONF_VARS["BE"]["unzip"]["list"]["file_pos"] = 3; // Note that this count starts at 0, so the 4th
element is 3
```

If everything is setup correctly for the listing the recognition of ".php" files will work correctly. To get the listing of extracted files to work properly some further steps are required.

The unzip configuration

Type the following command in the directory where you have your simple zip file:

```
unzip -o test.zip
```

If you get a output like the one below you are safe and won't need to change a thing:

```
Archive:  test.zip
 extracting: 1.html
 extracting: 2.html
 extracting: 3.html
 extracting: 4.html
```

You will need again to configure how many lines are before and after the listing of the extracted file. In our case the setup is as following:

```
$TYPO3_CONF_VARS["BE"]["unzip"]["unzip"]["pre_lines"] = 1;
$TYPO3_CONF_VARS["BE"]["unzip"]["unzip"]["post_lines"] = 0;
```

After this you will need again to set the split character by which the single parts of a line are recognized and the number of the element containing the filename. In our case this is as follows:

```
$TYPO3_CONF_VARS["BE"]["unzip"]["unzip"]["split_char"] = ":";
$TYPO3_CONF_VARS["BE"]["unzip"]["unzip"]["file_pos"] = 1;  // Note that counting begins at 0, so 1 is
the second element
```

After setting up those 8 values everything should work as expected.

Known problems

1. When unpacking a tar.gz or tar.bz2 files without overwriting other files, it will also show those files as extracted which already existed and weren't overwritten. With zip files just those files which really are extracted get shown.

To-Do list

- Add support for "rar" files
- Add support for other compression/decompression methods

Changelog

kb_unpack:

Version	Release Date	Changelog
0.0.0	?	• Initial release

0.0.1	?	• Made it CGL360 compliant. Hope to get reviewed.
0.0.2	2005-03-20, 16:00 CET	• Reindexed it with extdeveval
0.1.0	2006-04-14, 13:30 CET	• Addes support for filename with spaces inside. Thanks to Michiel Roos from netcreators for pointing this issue out. • Set extension state to stable.

kb_packman:

Version	Release Date	Changelog
0.2.0	2008-02-28 22:30 CET	• Initial release (Copy of kb_unpack 0.1.0). Added support for compressing directories and files into a zip-file.
0.2.1	2008-03-11 18:30 CET	• Corrected a bug: Extension only worked properly when DAM is installed. Now it works fine also without DAM (When configured correctly in the EM) - thx to Clemens Riccabona for reporting this bug.