

e_Version

Copyright Barry Keal 2006 Published under the GPL

This plugin is for plugin developers. They can add a link from their plugin's admin pages to allow users to check for any updates to the plugin and, if there are, then they can go to your site to get the details and download the update.

About

This plugin is designed to assist e107 site administrators with the regular updating of plugins. A Check for updates button is included somewhere on the plugins configuration pages. I prefer to use a menu on all my plugins so I make mine one of these options. The choice is yours.

When an admin wants to check to see if they have the latest version of their plugin then they click on the Check Updates button. This then runs the script "admin_vupdate.php", this script checks for a file called "e_update.php" and reads it in. This file contains the URL to your current version list in XML format, held on your server.

The data is retrieved from your site, compared with the current version of the plugin on the user's server and advises them if they have the latest version or if there is an update available. If the latter, then there is an optional link to the download page for the update.

There is also an optional link to your bugtracker or support forum for the plugin which is displayed for your user.

For your own support site there is a menu to display the current versions of your plugins or you can link to a full page listing the plugins and version histories.

Adding e_Version to your plugins.

There are several steps you need to go through to add e_Version to your plugin

1. Download and install e_version on your server and configure.
2. Create a list of your plugins using the plugin section of e_version
3. Create a link to the "check for updates" script admin_vupdate.php. This is most easily put into admin_menu.php along with the other menu options but is not compulsory. This needs to run the following script in your plugins directory - admin_vupdate.php
4. Copy admin_vupdate.php from e_version's distribution directory to the plugin's directory of the plugin you are enhancing.
5. Edit the e_version.php file so that the URL is pointing to the file eversion.xml in the xml directory of e_version on your web server. For example: http://www.yourdomain.com/e107_plugins/e_version/xml/eversion.xml
6. Copy e_version.php from the distribution directory to your plugin's directory after you have ensured that the URL points to the correct location on your server.
7. Add the eversion language folder to your plugin's language folder. You can create additional language files in the usual way if you wish. The admin_vupdate script will look in the languages/eversion/English.php file for the defines.
8. Ensure that the name specified in \$eplug_name in your plugin's plugin.php script is the same as the "name" field in e_version and that the version

number you store in `e_version` is the same as `$seplug_version` in `plugin.php` in the released version of your plugin. This is important, otherwise it won't work.

9. Check out this eversion plugin to see how I have done it.

E_Version Plugin Details

Create a new plugin entry

1. Plugin name : the name of the plugin in your plugin's `plugin.php` file (`$seplug_name`)
2. Plugin Title : the name displayed in the menu, main page and search.
3. Major, Minor and Beta release version numbers. These are compared with `$seplug_version` in `plugin.php`
4. Date of version : Date this release was issued.
5. Plugin Author : The name of the plugin author, normally you.
6. Revisions : Any version history details, this is shown to the user when they click to view the updates.
7. Developer Comments : This may be a todo list or whatever.
8. Path to Download : A link to the download entry on your site or another site (eg e107 coders)
9. Path to Support Forum or Bugtracker : If you have a support forum or bugtracker entry for this plugin then this link will take the user there.

Save the entry.

Version numbering

This is in the form `major.minor.beta`

For example 3.1.2 is major version 3, minor version 1 and release candidate (beta version) 2. A beta version of zero indicates it is a production version. This format is essential to the correct functioning of `e_version`. It detects that a beta version would be a higher value than a release version and compensates for this. For example is version 1.2.0 (a stable release) is available it will flag an update is downloadable even if the user has version 1.2.2 installed.

How it works

When you distribute your plugin the user will have a button called *Check for Updates*. When they click on it, it will go and fetch the file located at the URL specified in the `e_update.php` file in that plugin's directory. This points to YOUR server's `e_version/xml` directory not on the user's (see step 6 above). It will then check the version number (`$seplug_version`) specified in `plugin.php` and the name of the plugin (`$seplug_name`).

If there is an update available it will inform the user and give them a link to the plugin versions page on your site. This can then have a link to your downloads page which you create in the record for the plugin.

No version or plugin information is stored on the user's web server other than what is in plugin.php. All that is accessible from your web server is the txt file which contains serialized data on the plugins you have specified. This does not contain any data about the plugins you actually have installed.

RSS Feeds

This is still a little experimental but should provide RSS Feeds for your latest updates. Set up using the RSS plugin. You can remove this if you don't want it by taking out the RSS shortcode in the template.

Template

Limited support is given for defining the template for the display of the plugin versions.

Support

Support for this plugin is only at www.keal.me.uk