

Plugin Packaging

Purpose

This document outlines the procedure to package up your plugin so it is available from within the HomeSeer application.

Packaging

Plugins and their components are contained in a standard zip file. The zip file may be created on any Windows system. The root of the zip file should contain your plugin EXE file and any dependent DLL files and HTML files. Your zip file should be named using the following format:

pluginID_version.zip

For example, the sample plugin zip file for version 1.0.0.0 of the plugin would be named:

```
HomeSeerSamplePlugin_1-0-0-0.zip
```

On the target system, the files need to be copied to specific folders, this is accomplished with the file named install.txt. This file instructs HomeSeer on where to copy the included files. Here is the sample plugin install.txt file:

```
HSPI_HomeSeerSamplePlugin.exe,.,0
HSPI_HomeSeerSamplePlugin.exe.config,.,0
sample-blank.html,.\html\HomeSeerSamplePlugin,0
sample-guided-process.html,.\html\HomeSeerSamplePlugin,0
sample-plugin-feature.html,.\html\HomeSeerSamplePlugin,0
```

The first and second line copy the plugin EXE file and its config file to the HomeSeer root folder. Lines 3-5 copy the plugins HTML files to a new folder named "HomeSeerSamplePlugin" which HomeSeer will create in the HTML folder. This folder uses your plugin ID as the name of the folder. This is required.

If your plugin includes any dependent DLL files, they need to be copied to the bin folder. For example, if the sample plugin required a dll called "Newtonsoft.Json.dll", the following line would be added to your install.txt file:

```
Newtonsoft.Json.dll,.\bin\HomeSeerSamplePlugin,0
```

It is recommended that you name the folder in the bin folder with your plugin ID in order to avoid conflicts with other plugins.



Make sure you note the case of the files and paths in the install.txt file. While Windows does not care about the case, Linux does, and your plugin will fail to install or not run properly if the case in the install.txt file does not match the actual case of the file.

In order for your plugin to find your dependent DLL files, make sure you have this path entered in your plugin.exe.config file. For example, using the DLL above, the sample plugin config file would be named "HSPI_HomeSeerSamplePlugin.exe.config" and the contents would be:

```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.6.2"/>
  </startup>
  <runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
      <probing privatePath="bin/HomeSeerSamplePlugin;bin/homeseer"/>
    </assemblyBinding>
  </runtime>
</configuration>
```



Take note of your <probing privatePath> and make sure your path separator is "/". This allows the path to work on both Windows and Linux without providing a separate config file.

Once your plugin is ready for testing it may be published as Beta for users to test. To publish your plugin submit to the plugin developers portal. See the Plugin Portal doc for details.

See the [Install File Format](#) section for details on how to format the install.txt file.

