

Plugin Configuration

Plugins in version 3.1+ support the use of a **config.json** file, that defines basic configuration settings for the plugin.



Due to backwards compatibility with version 3.0, config.json files are not required but are **strongly recommended** as they may be required in future versions.

Configuration Format

Example

/plugins/my_plugin/config.json

```
{
  "version": "1.0.0",
  "name": "My Plugin Name",
  "description": "A plugin like no other!",
  "authors": [
    {
      "name": "Phillips Data, Inc.",
      "url": "http://www.blesta.com"
    }
  ]
}
```

Definition

Option	Description						
version	The version of the plugin. This should be a semantic version number.						
name	The name of the plugin. This may also be a language definition.						
description	A brief description of the plugin. This may also be a language definition in Blesta v4.9.0+.						
authors	An array of author objects. <table><tr><th>Option</th><th>Description</th></tr><tr><td>name</td><td>The name of the author. This may be a language definition</td></tr><tr><td>url</td><td>The URL to link to for the author.</td></tr></table>	Option	Description	name	The name of the author. This may be a language definition	url	The URL to link to for the author.
Option	Description						
name	The name of the author. This may be a language definition						
url	The URL to link to for the author.						
logo	The path within the plugin to the logo file. If not set, defaults to views/default/images/logo.png .						

Using the Configuration

To use the configuration file you must include it within the constructor of your plugin using the [loadConfig\(\)](#) method.

/plugins/my_plugin/my_plugin_plugin.php

```
<?php
class MyPluginPlugin extends Plugin {
    public function __construct() {
        $this->loadConfig(dirname(__FILE__) . DS . "config.json");
    }
}
?>
```

If your configuration uses any language definitions, be sure to include those before loading the config.

/plugins/my_plugin/my_plugin_plugin.php

```
<?php
class MyPluginPlugin extends Plugin {
    public function __construct() {
        Language::loadLang("my_plugin_plugin", null, dirname(__FILE__) . DS . "language" . DS);
        $this->loadConfig(dirname(__FILE__) . DS . "config.json");
    }
}
?>
```