

# Configuration

## Overview

Just as with components and modules, each plugin allows for its own set of configuration values that can be set via the administrative interface.

## Defining Options

Configuration options can be defined in the plugin's manifest XML file located in the plugin's directory.

```
/app
.. /plugins
.. .. /{PluginType}
.. .. .. /{PluginName}
.. .. .. .. {PluginName}.xml
```

The XML file's root element will have a child node of <config>. Fields are then added and grouped by fieldsets. These fieldsets correspond to the tabs located in the admin side when viewing the plugin's options.

```
<?xml version="1.0" encoding="utf-8"?>
<extension>
  <config>
    <fieldset
      name="greetings"
      label="PLG_HELLO_WORLD_CONFIG_GREETING_SETTINGS_LABEL"
      description="PLG_HELLO_WORLD_CONFIG_GREETING_SETTINGS_DESC"
    >
      <field
        name="greeting"
        type="text"
        label="PLG_HELLO_WORLD_FIELD_GREETING_LABEL"
        description="PLG_HELLO_WORLD_FIELD_GREETING_DESC"
        default=""
      />
    </fieldset>
  </config>
</extension>
```

It is good practice to use the plugin's language file to define all the appropriate strings.

### Retrieving Values

One may quickly retrieve the options for any plugin by calling the `params()` method on the Plugin facade or directly accessing the method on the underlying `HubzeroPluginLoader` class. This method accepts two arguments of the plugin type and plugin name and returns a `HubzeroConfigRegistry` object.

```
$params = Plugin::params('hello', 'world');  
  
echo $param->get('greeting');
```

Alternatively, all plugin instances should already have their params available upon instantiation.

```
<?php  
  
class plgHelloWorld extends HubzeroPluginPlugin  
{  
    public function onGreeting()  
    {  
        echo $this->params->get('greeting');  
    }  
}
```