

Guide to cPanel Plugins

Guide to cPanel Plugins

The dynamicui Files

Add Plugins (11.42 and earlier)

Add Plugins (11.44 and later)

Pluggable Statistics Modules

Uninstall Plugins

Introduction

cPanel plugins add new functionality to the cPanel interface. Third-party developers can add icons to the cPanel *Home* interface that link to new cPanel interfaces or to other files or locations.

Note:

- For information about WHM plugins, read our [Guide to WHM Plugins](#) documentation.
- Make certain that your applications respect user privacy. We **strongly** recommend that you anonymize any data that you collect for analysis. For more information about how cPanel, L.L.C. handles private data, read our [cPanel Analytics](#) documentation.

Plugin development workflow

When you create a cPanel plugin, we recommend that you use the following workflow:

Note:

You can also create a cPanel plugin with WHM's [cPanel Plugin File Generator](#) interface (*WHM >> Home >> Development >> cPanel Plugin File Generator*).

1. Create your custom application (the plugin's backend code).

Important:

cPanel's Paper Lantern interface incorporates the [RequireJS module loader](#). RequireJS may conflict with the [webpack module loader](#). We **strongly** recommend that you do not use webpack with any cPanel plugin or custom interface that you develop.

2. Create the plugin's interfaces.

Notes:

- For more information about how to create a new cPanel interface, read our [Create a New Paper Lantern Interface](#) and [Create a New Paper Lantern Interface in PHP](#) tutorials.
- For more information about how to customize the appearance of the cPanel interface, read our [Guide to cPanel Interface Customization and Branding](#) documentation.

3. Add the plugin to the cPanel interface. For the Paper Lantern theme, this requires that you [create an install.json file](#).
4. For plugins on servers that run cPanel & WHM version 11.42 and earlier, [write a plugin installation script](#).
5. **Optional:** Compress your plugin for distribution.
 - The compressed file contains the application source files, the plugin registration file, and the installation script.
 - We recommend that you compress your plugin as a tarball (`.tar.gz`) file. However, the `install_plugin` script supports `.tar.gz`, `.bz2`, and `.zip` files, and uncompressed directories.
6. For plugins on servers that run cPanel & WHM version 11.44 and later, [use the `/usr/local/cpanel/scripts/install_plugin` script to install the plugin](#).

Note:

The `/usr/local/cpanel/scripts/install_plugin` script does **not** extract plugin files for you. Instead, your installation script

cPanel Plugins

About cPanel Plugins

- [cPanel Plugins](#)

must perform this action.

Plugin files

cPanel interfaces can include HTML, [Template Toolkit](#), PHP, or CGI files.

- Store plugin interface files in the `/usr/local/cpanel/base/frontend/theme/` directory, where `theme` represents `paper_lantern` or another theme on the server.
- Access these files from `https://example.com:2083/frontend/theme/filename`, where:
 - `example.com` represents the domain or IP address.
 - `theme` represents `paper_lantern` or another theme on the server.
 - `filename` represents the interface file's name.

For example, if you create a custom *Support* interface in the `customsupport.html` file, use the following locations:

- Store the file in the following location on the server: `/usr/local/cpanel/base/frontend/paper_lantern/customsupport.html`
- Access this file at the following URL: `https://example.com:2083/frontend/paper_lantern/customsupport.html`

Note:

Additional resources to help you match custom interfaces to the cPanel interface exist in our [experimental User Interface Style Guide](#).

Installation scripts

Write any plugin installation scripts in your preferred programming language.

Plugin installation scripts must perform the following functions:

1. Verify that the system is compatible with the plugin. For example, confirm that the server uses a compatible version of cPanel & WHM.
2. Use the `wget` or `curl` commands to download the compressed plugin file.
3. Extract the compressed file's contents to the appropriate locations.
4. Perform the necessary tasks to add the plugin to the cPanel *Home* interface.
5. Remove any installation files that remain.

The `install_plugin` script

Important:

Due to the permissions that the `/usr/local/cpanel/scripts/install_plugin` script requires in order to access the necessary files, **only** the `root` user can run this script successfully.

cPanel & WHM version 11.44 introduced the `install_plugin` script, which you can use to install a plugin to a theme.

To use this script to install a plugin to cPanel, run the following command:

```
/usr/local/cpanel/scripts/install_plugin
plugin_file --theme theme_name
```

In this command, `plugin_file` represents the compressed plugin installation file (a `.tar.gz`, `.bzip`, or `.zip` file or an uncompressed directory), and `theme_name` represents `paper_lantern` or another theme on the server.

• E
a
c
h
c
P
a
n
e
l
t
h
e
m
e
h
a
s
d
i
f
f
e
r
e
n
t
r
e
q
u
i
r
e
m
e
n
t
s
f
o
r
c
P
a
n
e
l
p
l
u
g
i
n
s
.

Compati
ble with:

• c
P
a
n
e
l
1
1
.
3
2
+

Notes:

- When you run this script, it **only** installs the plugin to **one** theme. To install the plugin to a second theme, you **must** run the script again with the correct options.
- If you do not specify the `--theme` option and a theme, the script attempts to install the plugin to the Paper Lantern interface with the `install.json` method. If you do not specify a theme and the compressed plugin file does not contain an `install.json` file, the installation will **fail**.
- The `/usr/local/cpanel/scripts/install_plugin` script does **not** extract plugin files for you. Instead, your installation script **must** perform this action.

Related Documentation

- [Guide to cPanel Plugins - Add Plugins to Legacy cPanel Versions](#)
—
cPanel & WHM version 11.4.2 and earlier requires you to manually add items to the dynamic icui files.

- [Guide to cPanel Variables](#)

—
The Live API system and dynamic files can access cPanel's global variables.

- [Tutorial - Create a New Paper Lantern Interface in PHP](#)

—
This tutorial uses PHP to create a new cPanel interface for the Paper Lantern theme.

- [Tutorial - Add a Link to the cPanel Interface](#)

— This tutorial adds an icon that links to an external location to the cPanel interface.

- [Guide to cPanel Plugins - Uninstall Plugins](#)

— To uninstall plugins from cPanel & WHM, choose the uninstallation process according to your system's version.