

MultiPHP INI Editor for cPanel

(cPanel >> Home >> Software >> MultiPHP INI Editor)

Important:

cPanel's *MultiPHP INI Editor* interface **only** exists on systems that run EasyApache 4 and use the Paper Lantern theme.

Overview

This interface allows you to configure your PHP settings.

Note:

WHM users can view information about the server's PHP INI settings in WHM's *MultiPHP INI Editor* interface (WHM >> Home >> Software >> MultiPHP INI Editor).

MultiPHP INI Editor

Select the location that you wish to configure from the menu. You can edit either the account's home directory or the domain's document root. The PHP directives will appear.

Select a tab to view more information about that section of the *MultiPHP INI Editor* interface.

Basic Mode Editor Mode

The *Basic Mode* section of this interface loads directive values from the `php.ini` file for your selected domain or home directory. Make your changes to the desired directives and click *Apply*. The system saves changes to the `php.ini` file, the `user.ini` file, and your `.htaccess` file.

Notes:

- For security purposes, we **strongly** recommend that you use the cPanel-provided default values.
- The availability of directives depends on your version of PHP. If a directive does **not** appear in your interface, your version of PHP does **not** support it.
- The system **only** writes a directive to an `.ini` or `.htaccess` file if that file supports the directive. Your `php.ini`, `.user.ini`, and `.htaccess` files may not be identical. For more information about which files support which directives, read PHP's [documentation](#).

Important:

If you use the DSO PHP handler and you change your PHP version to a different major PHP version, the system will **not** read your configuration changes. For example, if you change your PHP version from PHP 5 to PHP 7.

To correct this issue, you **must** edit and save your PHP configuration again in the interface. The system will then load the `.htaccess` directives correctly.

In This Document

Related Documentation

- [PHP](#)
- [PHP PEAR Packages](#)
- [MultiPHP INI Editor for cPanel](#)
- [MultiPHP Manager for cPanel](#)

For Hosting Providers

- [EasyApache 4 Change Log 2016](#)
- [Scripts and Scripting Languages FAQ](#)
- [Apache vs PHP-FPM Bottleneck with Child Processes](#)
- [The rebuild_phpconf Script](#)
- [Configure PHP and suEXEC](#)

PHP directive	Description	cPanel default	PHP default
---------------	-------------	----------------	-------------

<i>allow_url_fopen</i>	<p>Select whether PHP scripts can use URL-aware fopen wrappers.</p> <div data-bbox="386 310 591 718" style="border: 1px solid orange; padding: 5px;"><p>Note: The system only displays this directive if you use the suPHP PHP handler.</p></div>	<i>Disabled</i>	<i>Enabled</i>
<i>allow_url_include</i>	<p>Select whether PHP scripts can use URL-aware fopen wrappers with the following functions:</p> <ul style="list-style-type: none">• <code>include()</code>• <code>include_once()</code>• <code>require()</code>• <code>require_once()</code> <div data-bbox="386 1251 591 1659" style="border: 1px solid orange; padding: 5px;"><p>Note: The system only displays this directive if you use the suPHP PHP handler.</p></div>	<i>Disabled</i>	<i>Disabled</i>

<i>asp_tags</i>	<p>Select whether PHP scripts can use ASP-like tags in addition to PHP tags.</p> <div style="border: 1px solid orange; padding: 5px;"><p>Note: This includes variable-value printing shorthand (for example, <code><%= \$value %></code>).</p></div>	<i>Disabled</i>	<i>Disabled</i>
<i>display_errors</i>	<p>Select whether PHP displays errors as output or hides them from the user.</p>	<i>Disabled</i>	<i>Enabled</i>

enable_dl

Select whether scripts can call the `d1` function. This function dynamically loads a PHP extension at runtime.

Disabled

Enabled

Warning

:
If you set this directive to *Enable*, it poses security risks to your server. We **strongly** recommend that you select *Disabled* unless you **absolutely** require the `d1` function.

Note:

The system **only** displays this directive if you use the `suhp` PHP handler.

<i>file_uploads</i>	<p>Select whether your PHP scripts can upload HTTP files.</p> <div style="border: 1px solid orange; padding: 5px; margin: 10px 0;"> <p>Note: The system o nly displays this directive if you use the suPHP P PHP handler.</p> </div>	<i>Enabled</i>	<i>Enabled</i>
<i>magic_quotes_gpc</i>	Select whether PHP automatically escapes certain characters with a backslash (\).	<i>Enabled</i>	<i>Enabled</i>
<i>max_execution_time</i>	Enter the maximum amount of time, in seconds, that your server allows a script to run before your server terminates it. This setting limits the amount of memory that poorly-written scripts may use.	30	30
<i>max_input_time</i>	Enter the maximum amount of time, in seconds, that your server allows a script to parse input data.	60	-1

<i>max_input_vars</i>	Enter the maximum number of input variables per request. Use this directive to deter Denial of Service (DoS) attacks that use hash collisions on input variable names.	1000	1000
<i>memory_limit</i>	<p>Enter the maximum amount of memory that scripts can allocate, in bytes. This setting helps to prevent poorly-written scripts that may use too much memory.</p> <p>cPanel & WHM sets the default value based on the system's memory at the time of install.</p> <div data-bbox="386 1226 591 1604" style="border: 1px solid orange; padding: 5px; margin-top: 10px;"> <p>Note: Use the character M to define the limit in Megabytes.</p> </div>	<p>cPanel & WHM adjusts this setting at the time of install, based on system memory:</p> <ul style="list-style-type: none"> • 1 GB — 32M • 2-4GB — 64M • Over 4 GB — 128M 	128M
<i>post_max_size</i>		8M	8M

Enter the maximum size of post data that the system allows, in bytes. This setting affects file uploads. Set this value to 0 to allow unlimited post sizes.

Notes:

- We strongly recommend that you set this value larger than the upload_max_filesize value and smaller than the memory_limit value.
- Use the character M to define the limit in Mega bytes.

<i>register_globals</i>	Select whether GET or POST input converts into variables that the script can use.	<i>Disabled</i>	<i>Disabled</i>
<i>safe_mode</i>	Select whether to enable safe mode. Safe mode increases security for PHP scripts and enables user, group, and environment checks and controls. <div style="border: 1px solid orange; padding: 5px; margin: 10px 0;"> <p>Note: The system o nly displ ays this directive if you use the suPH P PHP handler.</p> </div>	<i>Disabled</i>	<i>Disabled</i>
<i>session.save_path</i>	Enter the directory in which your server stores the files that PHP creates.	NULL	NULL

<code>upload_max_filesize</code>	Enter the maximum file size for an upload, in bytes. Note: Use the character M to define the limit in Megabytes.	2M	2M
<code>zlib.output_compression</code>	Select whether to transparently compress pages when the browser sends an Accept-Encoding: gzip or deflate header.	<i>Disabled</i>	<i>Disabled</i>

Select the location that you wish to configure from the menu. You can edit either the user account's home directory or the domain's document root. The PHP directives will appear.

Warnings:

- You **must** exercise extreme caution when you make changes in this interface. Errors could result in non-functional PHP scripts. **Only** advanced users should use this interface.
- We restrict the following words and you **cannot** use them as directives: `null`, `yes`, `no`, `true`, `false`, `on`, `off`, and `none`. Directives also **cannot** contain any the following special characters: `?{ } | & ~ ! [() ^ "`
- The following values set the directive value to "": `null`, `off`, `no`, and `false`.
- The following values set the directive value to 1: `on`, `yes`, and `true`.

The *Editor Mode* section of this interface loads the contents of the `php.ini` file for your selected domain, if it exists. If the `php.ini` file does **not** exist, the interface loads a blank editor so that you can create your own file. Make your changes to the desired directives and click *Save*.

The system saves changes to both the `php.ini` file in the *Path* section of the interface and the `.user.ini` file. The system also saves these changes to your `.htaccess` file. For example, the location of your file may be the `/home/allthethings/public_html/php.ini` file. The interface attempts a rudimentary validation of directive names. This does **not** guarantee that your PHP configuration will function. To reset your PHP settings to default values, delete the domain's or home directory's `php.ini` file.

Note:

The system **only** writes a directive to an `.ini` or `.htaccess` file if that file supports the directive. Your `php.ini`, `.user.ini`, and `.htaccess` files may not be identical. For

more information about which files support which directives, read the [PHP documentation](#).

Important:

If you use the DSO PHP handler, the system will **not** read your configuration changes if you change your PHP version to a different major PHP version. For example, you change your PHP version from PHP 5 to PHP 7.

To correct this issue, you **must** edit and save your PHP configuration again in the interface. The system will then load the `.htaccess` directives correctly.

For more information about a directive, search the [php.net](#) documentation.