How to Set Up (PHP) Composer

> ⊘ **Warning:**
>
> cPanel support does **not** provide assistance with the instructions in this document.

Overview

Composer is a dependency and package manager for PHP. For more information about Composer, read the Composer documentation. You can search for packages to download at the packagist.org website.

> ⚠ **Notes:**
>
> - Your cPanel account **must** possess SSH access in order to use Composer.
> - We added Composer in cPanel & WHM version 58.

How to set up Composer

In order to use Composer, you **must** install several PHP modules. You must also verify that you have properly configured your server's `php.ini` file to allow Composer to run.

By default, when you call Composer, it **only** uses the server's `php.ini` file. The system stores this file in the `/opt/cpanel/ea-*/root/etc/php.ini` location.

Required PHP modules

> ⚠ **Note:**
>
> You **must** perform these steps as the `root` user.

To use Composer, you **must** enable several PHP modules in EasyApache. For more information about EasyApache 4, read our EasyApache documentation.

PHP Module	Description
Phar	This module provides a way to store entire PHP applications into a single "phar" file (PHP Archive) for easy distribution and installation. > ⚠ **Note:** > > EasyApache4 installs this module by default.
Iconv	This module provides a Character Set Conversion > ⚠ **Note:** > > If you install this module, the `MBstring` module is **not** required.

⚠

Mbstring	This module provides enhanced Chinese, Japanese, and other language support.
	⚠️ **Note:** If you install this module, the `Iconv` module is **not** required.

Required PHP directives

You **must** update the following PHP directives in the `php.ini` file:

> ⚠️ **Note:**
>
> For more information about how to edit your `php.ini` file, read our How to Edit Your `php.ini` File documentation.

Directive	Required value
`allow_url_fopen`	On
`zend.detect_unicode`	Off

> ⚠️ **Note:**
>
> For servers that use PHP version 5.3 and earlier, use the `detect_unicode` directive instead of the `zend.detect_unicode` directive.

Servers that use Suhosin

If you use Suhosin, you **must** add the Phar PHP module to the Suhosin whitelist in the server's `php.ini` file. Add the following line to you server's `php.ini` file:

```
suhosin.executor.include.whitelist=phar
```

> ⚠️ **Note:**
>
> For more information about how to edit your `php.ini` file, read our How to Edit Your `php.ini` File documentation.

What if I cannot edit the php.ini file?

If you cannot make changes to the `php.ini` file, you **must** manually run the following command each time you run Composer:

- With Suhosin:

```
php -d allow_url_fopen=1 -d zend.detect_unicode=0 -d suhosin.executor.include.whitelist=phar $(which composer)
```

- Without Suhosin:

```
php -d allow_url_fopen=1 -d zend.detect_unicode=0 $(which composer)
```

If you use Composer often, you can create a Bash alias to perform this step for you. If you add an alias, you can run the `composer` command without any additional flags or options. Add the following line to your `.bashrc` file to create this alias each time that you log in to the server :

- With Suhosin:

```
alias composer="php -d allow_url_fopen=1 -d zend.detect_unicode=0 -d suhosin.executor.include.whitelist=phar $(which composer)"
```

- Without Suhosin:

```
alias composer="php -d allow_url_fopen=1 -d zend.detect_unicode=0 $(which composer)"
```

Troubleshoot Composer

Command not found

If you receive the `command not found` error message, you may need to add the proper path to your account's `.bash_profile` file.

Add the following lines to your account's `.bash_profile` file:

```
source /etc/profile.d/cpanel-php-composer.sh
```

If you experience problems with the `source` option, add the `PATH` option to the `.bash_profile` file instead:

```
PATH=$PATH:/opt/cpanel/composer/bin
```

> ⚠️ **Note:**
>
> You **must** log into the server again for any changes to the `.bash_profile` file to take effect.

The Composer self-diagnostic tool

Composer contains a self-diagnostic tool to ensure that you have properly configured PHP.

Run the diagnostic tool with the following command:

```
composer diagnose
```

The `Checking platform setting` section provides feedback about your server's PHP configuration. It will also inform you of any PHP configuration failures.

The following example demonstrates this section:

```
 $ composer diagnose
Checking platform settings: OK
Checking git settings: OK
Checking http connectivity to packagist: Warning: Accessing packagist.org over http which is an insecure
protocol.
OK
Checking https connectivity to packagist: OK
Checking github.com rate limit: OK
Checking disk free space: OK
Checking pubkeys: FAIL
Missing pubkey for tags verification
Missing pubkey for dev verification
Run composer self-update --update-keys to set them up
Checking composer version: FAIL
You are not running the latest stable version, run `composer self-update` to update (1.1.1 => 1.1.2)
```

Alternative diagnostic tools

If you require an alternate diagnostic tool, use the diagnostic tool from Composer's Github account. Run the following command:

```
curl https://raw.githubusercontent.com/composer/getcomposer.org/master/web/installer | php -- --check
```

This utility only checks the server's PHP configuration, as in the following example:

```
# curl https://raw.githubusercontent.com/composer/getcomposer.org/master/web/installer | php -- --check
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  286k  100  286k    0     0   466k      0 --:--:-- --:--:-- --:--:--  1002k
All settings correct for using Composer
#
```

Additional documentation

- How to Set Up (PHP) Composer
- How to Build and Install Custom RPMs
- Third-Party Software End Of Life Policy
- Additional Security Software
- CVE-2016-3714 ImageMagick