

# PHP

- [PHP version setting for DirectAdmin web hosting](#)
- [How to use lsPHP on my DirectAdmin webspace](#)
- [How to use suPHP on my webspace](#)
- [Set custom PHP version per directory](#)

# PHP version setting for DirectAdmin web hosting

1. Sign in at <https://admin.dotroll.com/> with your username and password!
2. In the top menubar, click the **Services / My Services** menu!
3. From your services, click the DirectAdmin web hosting!
4. Scroll down, in the One Click Menu on the left, and click on the DirectAdmin option, which will take you to the DirectAdmin interface!
5. Click the **“Select PHP version”** option in the **Extra Features** block!
6. **Default setting is native 7.3 version**, if you change it to another version, you can turn different PHP modules on and off, as you can see in the picture.

php-version-allitas-directadmin-1

7. Above, if you click on the **Options** button, **you can edit different PHP settings**. If you want to change the modules, you can set the version to another (default is 7.3 native).

Most frequently used parameters:

- **upload\_max\_filesize** - maximum file size that can be uploaded to a website
- **memory\_limit** - how much memory to use for a given PHP query
- **max\_execution\_time** - how long a PHP script should run.

php-version-allitas-directadmin-2

The PHP version you set for the main domain will be valid for the hosting, but you can specify different PHP settings, the steps for which can be found in [this article](#).

# How to use lsPHP on my DirectAdmin webspace

## WHAT IS LSPHP?

lsphp - PHP + LSAPI.

## WHAT IS LSAPI?

LiteSpeed Server Application Programming Interface (LSAPI) is designed specifically for seamless, optimized communication between LiteSpeed Web Server and third party web applications. The lsapi is the fastest and most reliable way to serve PHP pages. It has a low memory footprint and understands PHP directives from .htaccess files.

If you don't want to use the default PHP version under on your domain name, you need to create a .htaccess file in a particular domain name assigned to a folder (document\_root) will be placed into one of the following lines. After setting recursively, all sub-directories will be applied to the setting.

PHP version 4.4:

```
AddHandler application/x-httpd-lsphp44 .php .php5 .php4 .php3
```

PHP version 5.1:

```
AddHandler application/x-httpd-lsphp51 .php .php5
```

PHP version 5.2:

```
AddHandler application/x-httpd-lsphp52 .php .php5
```

PHP version 5.3:

```
AddHandler application/x-httpd-lsphp53 .php .php5
```

PHP version 5.4:

```
AddHandler application/x-httpd-lsphp54 .php .php5
```

PHP version 5.5:

```
AddHandler application/x-httpd-lsphp55 .php .php5
```

PHP version 5.6:

```
AddHandler application/x-httpd-lsphp56 .php .php5
```

PHP version 7.0:

```
AddHandler application/x-httpd-  
lsphp70 .php .php5
```

PHP version 7.1:

```
AddHandler application/x-httpd-lsphp71 .php .php5
```

PHP version 7.2:

```
AddHandler application/x-httpd-lsphp72 .php .php5
```

PHP version 7.3:

```
AddHandler application/x-httpd-lsphp73 .php .php5
```

PHP version 7.4:

```
AddHandler application/x-httpd-lsphp74 .php .php5
```

PHP version 8.0:

```
AddHandler application/x-httpd-lsphp80 .php .php5
```

PHP version 8.1:

```
AddHandler application/x-httpd-lsphp81 .php .php5
```

The module enabled per PHP version is a global setting, but you can change your php.ini settings with a .php.ini file placement. Starting with PHP 5.4 version, you can put a .user.ini file, each of which can override PHP settings for that folder (and its subdirectories).

# How to use suPHP on my webspace

## WHAT IS THE SUPHP?

suPHP is a tool for executing PHP scripts with the permissions of their owners. It consists of an Apache module (mod\_suphp) and a setuid root binary (suphp) that is called by the Apache module to change the uid of the process executing the PHP interpreter.

Works with the PHP versions available in the PHP version option on the cPanel interface.

If you don't want to use the default PHP version under on your domain name, you need to create a .htaccess file in a particular domain name assigned to a folder (document\_root) will be placed into one of the following lines. After setting recursively, all sub-directories will be applied to the setting.

PHP version 4.4:

```
AddHandler application/x-httpd-suphp44 .php .php5 .php4 .php3
```

PHP version 5.1:

```
AddHandler application/x-httpd-suphp51 .php .php5
```

PHP version 5.2:

```
AddHandler application/x-httpd-suphp52 .php .php5
```

PHP version 5.3:

```
AddHandler application/x-httpd-suphp53 .php .php5
```

PHP version 5.4:

```
AddHandler application/x-httpd-suphp54 .php .php5
```

PHP version 5.5:

```
AddHandler application/x-httpd-suphp55 .php .php5
```

PHP version 5.6:

```
AddHandler application/x-httpd-suphp56 .php .php5
```

PHP version 7.0:

```
AddHandler application/x-httpd-suphp70 .php .php5
```

PHP version 7.1:

```
AddHandler application/x-httpd-suphp71 .php .php5
```

PHP version 7.2:

```
AddHandler application/x-httpd-suphp72 .php .php5
```

PHP version 7.3:

```
AddHandler application/x-httpd-suphp73 .php .php5
```

PHP version 7.4:

```
AddHandler application/x-httpd-suphp74 .php .php5
```

PHP version 8.0:

```
AddHandler application/x-httpd-suphp80 .php .php5
```

PHP version 8.1:

```
AddHandler application/x-httpd-suphp81 .php .php5
```

The module enabled per PHP version is a global setting, but you can change your php.ini settings with a .php.ini file placement. Starting with PHP 5.4 version, you can put a .user.ini file, each of which can override PHP settings for that folder (and its subdirectories).

# Set custom PHP version per directory

On the cPanel interface you can set the default PHP version on the Select PHP version menu.

Currently the following PHP versions are available:

- PHP 4.4
- PHP 5.1
- PHP 5.2
- PHP 5.3
- PHP 5.4
- PHP 5.5
- PHP 5.6
- PHP 7.0
- PHP 7.1
- PHP 7.2
- PHP 7.3
- PHP 7.4
- PHP 8.0
- PHP 8.1
- PHP 8.2

If you are not using the native PHP version you can turn it on the available PHP modules, and you can modify some PHP settings directly in php.ini file with every PHP version. Note: In the PHP modules, it is important to pay attention to is not enough to checking the check boxes, you also need to save the settings on the page!

If you don't want to use the default PHP version under on your domain name, you need to create a .htaccess file in a particular domain name assigned to a folder (document\_root) will be placed into one of the following lines. After setting recursively, all sub-directories will be applied to the setting.

PHP version 4.4:

```
AddHandler application/x-httpd-lsphp44 .php .php5 .php4 .php3
```

PHP version 5.1:

```
AddHandler application/x-httpd-lsphp51 .php .php5
```

PHP version 5.2:

```
AddHandler application/x-httpd-lsphp52 .php .php5
```

PHP version 5.3:

```
AddHandler application/x-httpd-lsphp53 .php .php5
```

PHP version 5.4:

```
AddHandler application/x-httpd-lsphp54 .php .php5
```

PHP version 5.5:

```
AddHandler application/x-httpd-lsphp55 .php .php5
```

PHP version 5.6:

```
AddHandler application/x-httpd-lsphp56 .php .php5
```

PHP version 7.0:

```
AddHandler application/x-httpd-lsphp70 .php .php5
```

PHP version 7.1:

```
AddHandler application/x-httpd-lsphp71 .php .php5
```

PHP version 7.2:

```
AddHandler application/x-httpd-lsphp72 .php .php5
```

PHP version 7.3:

```
AddHandler application/x-httpd-lsphp73 .php .php5
```

PHP version 7.4:

```
AddHandler application/x-httpd-lsphp74 .php .php5
```

PHP version 8.0:

```
AddHandler application/x-httpd-lsphp80 .php .php5
```

PHP version 8.1:

```
AddHandler application/x-httpd-lsphp81 .php .php5
```

PHP version 8.2:

```
AddHandler application/x-httpd-lsphp82 .php .php5
```

The module enabled per PHP version is a global setting, but you can change your php.ini settings with a .php.ini file placement. Starting with PHP 5.4 version, you can put a .user.ini file, each of which can override PHP settings for that folder (and its subdirectories).