> **Bear in mind that this documentation is for the last released version. If you use an older version, you must refer to the README.md file inside the plugin archive.**

i-MSCP PhpSwitcher plugin v5.0.0

Provides additional PHP versions for customers.

Requirements

- One of the supported distributions
- i-MSCP Serie ≥ 1.5.3
- The Fcgid or the FPM i-MSCP httpd server implementation

Supported distributions

- Debian Jessie 8.x, Debian Stretch 9.x, Debian Buster 10.x
- Ubuntu Xenial Xerus 16.04, Ubuntu Bionic Beaver 18.04
- Devuan Jessie 1.x, Devuan ASCII 2.x

Recommendations

Recommended use of this plugin is with the i-MSCP FPM httpd server in **UDS** mode, and with the **per_site** PHP configuration level.

Installation

1. Be sure that all requirements are met
2. Upload and install the plugin through the plugin management interface
3. Compile, install and/or register your PHP versions

Update

1. Read the UPDATE.md file
2. Be sure that all requirements are met
3. Upload the plugin through the plugin management interface
4. Compile, install and/or register your PHP versions

Glossary

In the documentation below, the following variables are used:

- %version% : One of available PHP version such as 7.1, 7.2, 7.3, 7.4 …
- %ymd% : The date of the day such as 20190831
- %installdir% : Installation directory such as /opt/phpswitcher/20170306

Compiled PHP versions

This plugin provides the `php_compiler.pl` script that can, compile and install additional PHP versions in one step. It can also register them in the Phpswitcher plugin.

For example, if you want to compile and install all additional PHP versions, you can execute the following command:

```
# perl /var/www/imscp/gui/plugins/PhpSwitcher/PhpCompiler/php_compiler.pl
```

or if you want to install specific PHP versions:

```
# perl /var/www/imscp/gui/plugins/PhpSwitcher/PhpCompiler/php_compiler.pl
%version% %version%
```

You can also automatically register the PHP versions in the PhpSwitcher plugin by passing the `--register` option. For instance:

```
# perl /var/www/imscp/gui/plugins/PhpSwitcher/PhpCompiler/php_compiler.pl --
register %version% %version%
```

Doing this avoid the need to register them manually through the PhpSwitcher plugin administration interface.

Supported PHP versions

The supported PHP versions are the last that were available when this plugin version has been released. This means that by default, the versions provided by the `php_compiler.pl` script can be lower than the last that were released on the PHP site.

Supported PHP versions are: `5.3` up-to `7.4`

Forcing the `php_compiler.pl` script to download latest PHP sources

You can ask the `php_compiler.pl` script to download latest PHP sources by passing the `--force-last` option. However, you must bear in mind that some patches might not longer apply, in which case you should report us the failure message by creating a new issue on our bug tracker.

Download directory

By default, the PHP sources are downloaded in the `/usr/local/src/phpswitcher` directory. You can override the default download directory by passing the `--download-dir` option to the `php_compiler.pl` script.

Build directory

By default, the compiled PHP versions are build in the `/usr/local/src/phpswitcher/php%version%` directory. You can override the default build directory by passing the `--build-dir` option to the `php_compiler.pl` script.

Installation directory

By default, the compiled PHP versions are installed in the
`/opt/phpswitcher/%ymd%/php%version%` directory. You can override the default installation
directory by passing the `--install-dir` option to the `php_compiler.pl` script.

Changes made on PHP sources

A set of patches is applied on PHP sources before configuring them. The patches include the following
changes:

- Multiarch support
- Support for latest OpenSSL 1.1.x version
- Any patch that fix a bug, a security issue or a FTBFS issue

Most of the patches were pulled from the Debian PHP team git repository and modified when needed,
while some other were created to resolve FTBFS issues. Patches which were not pulled from Debian
PHP team git repository have the `nxw-` prefix in their names.

PHP extensions

PHP extensions are the same that are provided by PHP versions from the Debian PHP team. Most are
compiled as shared module.

PHP build dependencies (since version 5.0.0)

The `php_compiler.pl` script install the build dependencies for you by creating a specific Debian
dependency package for each PHP version. These packages are purged after a successfull build.

PHP runtime dependencies (since version 5.0.0)

The `php_compiler.pl` script install the PHP runtime dependencies for you by creating a specific
Debian runtime dependency package for each PHP version. These packages are purged only when
you uninstall the plugin.

Build environment (since version 5.0.0)

Build perations are performed in a chroot environment by default. This allows to setup a clean build
environment, and this also avoid to pollute the production system with PHP build dependencies.

You can switch back to the historical behavior by passing the `--no-chroot` option to the
`php_compiler.pl` script.

Parallel Execution (GNU make)

For faster compilation, the parallel execution feature provided by GNU make is enabled. This feature
allows to execute many recipes simultaneously. By default, the number of parallel jobs is set
according the number of CPU core available on your system + 1. You can still set the value manually
by passing the `--parallel-jobs` option to the `php_compiler.pl` script.

See Parallel Execution for further details about this feature.

Installation layout

For all PHP versions provided by the PHP compiler, the following installation layout applies:

Common to all PHP SAPIs

- Main php.ini file: /opt/phpswitcher/%ymd%/php%version%/etc/php/php.ini
- Additional *.ini files: /opt/phpswitcher/%ymd%/php%version%/etc/php/conf.d

PHP cli SAPI

- PHP CLI binary: /opt/phpswitcher/%ymd%/php%version%/bin/php

PHP cgi SAPI

- PHP FastCGI binary: /opt/phpswitcher/%ymd%/php%version%/bin/php-cgi
- User php.ini files: /var/www/fcgi/%domain%/php%version%/php.ini

PHP fpm SAPI

- PHP-FPM binary: /opt/phpswitcher/%ymd%/php%version%/sbin/psw%version%-fpm
- PHP-FPM configuration file: /opt/phpswitcher/%ymd%/php%Version%/etc/php-fpm.conf
- PHP-FPM instance pid file: /var/run/phpswitcher/psw%version%-fpm.pid
- PHP-FPM log file: /var/log/phpswitcher/psw%version%-fpm.log
- PHP-FPM pool configuration files: /opt/phpswitcher/%ymd%/php%Version%/etc/php-fpm.d/%domain%.conf
- PHP-FPM reopenlogs script: /usr/local/lib/phpswitcher/psw%version%-fpm-reopenlogs
- PHP-FPM sysvinit script: /etc/init.d/psw%version%-fpm
- PHP-FPM tmpfile: /etc/tmpfiles.d/psw%version%-fpm.conf

As you can see, there is a specific naming convention for PHP-FPM. This is needed to avoid conflict with packaged PHP versions.

Note that the `psw` prefix stands for `PhpSwitcher`.

Static paths (since version 5.0.0)

For convenience, the `php_compiler.pl` script also creates symlinks to the installation directories of various PHP versions, which allows the sysadmin to access the PHP binaries always by the same paths (e.g. in CRONTAB(5) files).

These symlinks are created in the `/opt/phpswitcher/static` directory, either at the installation stage, or at the registration stage. If you do not want these symlinks, you can pass the `--no-static-paths` option to the `php_compiler.pl` script.

Note that when the symlinks are available, they will be used for registration process in place of the default paths.

Packaged PHP versions (since version 5.0.0)

The `php_compiler.pl` script also make possible to register packaged PHP versions in the PhpSwitcher plugin. Packaged PHP versions are either those provided by the Debian PHP team, or the Ondřej Surý repositories. To register the packaged PHP versions, you need to execute the following

command:

```
# perl /var/www/imscp/gui/plugins/PhpSwitcher/PhpCompiler/php_compiler.pl --
register --packaged
```

When the --packaged option is passed-in, the php_compiler.pl script will favor registration of packaged PHP versions rather than compiled ones. In other words, for a given PHP version, the PHP compiler will fallback to a compiled PHP version only if there is no packaged PHP version available.

If you only want work with packaged PHP versions, you can also pass the --packaged-only option:

```
# perl /var/www/imscp/gui/plugins/PhpSwitcher/PhpCompiler/php_compiler.pl --
register --packaged --packaged-only
```

With this option, only the packaged PHP versions will be considered, regardless of their availalibity. If no packaged PHP version is available, nothing will happend, that is, no fallback to a compiled PHP version will be made.

You can of course still specify the PHP versions for which you want to operate on, and you can also change the default directory, where the PHP compiler is looking for the compiled PHP versions, by passing the --install-dir option. For instance:

```
# perl /var/www/imscp/gui/plugins/PhpSwitcher/PhpCompiler/php_compiler.pl --
register --packaged --install-dir /opt/my/install/directory 7.1 7.3 7.4
```

Note that for a packaged PHP version, the plugin backend will skip generation of most configuration files as those are already provided by the package. You need also remember that for a packaged PHP version, the FPM service name is php%version%-fpm instead of psw%version%-fpm.

Finally, you need keep in mind that for the php_compiler.pl script, the packaged PHP versions are considered only when all requirements are met, that is, when for a given PHP version, the CGI sapi, FPM sapi, and the PEAR distribution are all installed on the system. For instance, if you want to register the PHP 7.3 packaged version, you must first make sure that all required distribution packages are installed:

```
# apt-get update
# apt-get install php7.3-cli php7.3-cgi php7.3-fpm php-pear
```

PEAR packages

For a compiled PHP version

Each compiled PHP version comes with its own PEAR distribution which is installed under the /opt/phpswitcher/%ymd%/php%version%/share/pear directory.

You shouldn't try to include a PEAR library from another location without knowing what you are doing because doing this could lead to an unexpected behavior.

If you want to install additional PEAR packages, you must install them using the pear command provided by the compiled PHP version for which you want act. For instance:

```
# cd /opt/phpswitcher/%ymd%/php%Version%/bin
# ./pear install Net_SMTP
```

For a packaged PHP version

For a packaged PHP version, you just need to install the distribution package:

```
# apt-get update
# apt-get install php-net-smtp
```

or if there is none, you can process as follows:

```
# /usr/bin/pear install Net_SMTP
```

PHP extensions

For a compiled PHP version

For convenience, most of PHP extensions are compiled as shared modules. When a compiled PHP version is being installed, a specific INI file that enable most of available PHP extensions is created. By default, this file is located at
`/opt/phpswitcher/%ymd%/php%version%/etc/php/conf.d/modules.ini`.

Here, a single INI file is involved. This is not like with a packaged PHP version where there is one INI file per extension.

To enable/disable a specific PHP extension, you must just edit the modules.ini INI file and once done, restart the Web server or PHP-FPM instance, depending on the i-MSCP httpd server implementation in use.

For a packaged PHP version

For a packaged PHP version, you can enable/disable the PHP extensions using the phpenmod/phpdismod command provided by your distribution.

PECL extensions

For a compiled PHP version

If you want to install a PECL extension, you must not invoke the `pecl` or `phpize` commands that are provided by your distribution. Instead, you must invoke those that are provided by the compiled PHP version for which you want act.

For instance, if you want to install the `Imagick` PECL extension you can process as follows:

```
# apt-get update
# apt-get -y install libmagickwand-dev imagemagick
# /opt/phpswitcher/%ymd%/php%version%/bin/pecl install imagick
# echo 'extension = imagick.so' >
/opt/phpswitcher/%ymd%/php%version%/etc/php/conf.d/imagick.ini
```

Finally, you must restart the Web server or PHP-FPM instance, depending on the i-MSCP httpd server implementation in use.

For a packaged PHP version

For a packaged PHP version, you just need to install the distribution package:

```
# apt-get update
# apt-get install php%version%-imagick
```

or if there is none, you can process as follows:

```
# apt-get update
# apt-get -y install php%version%-dev libmagickwand-dev imagemagick
# update-alternatives --set php /usr/bin/php%version%
# update-alternatives --set phpize /usr/bin/phpize%version%
# update-alternatives --set php-config /usr/bin/php-config%version%
# pecl install imagick
# echo 'extension = imagick.so' > /etc/php/%version%/mods-
available/imagick.ini
# phpenmod -v %version% imagick
```

Finally, you must restart the Web server, or PHP-FPM instance, depending on the i-MSCP httpd server implementation in use.

You can uninstall the PECL extension by following the same procedure, replacing the pecl `install` command by the pecl `uninstall` command. However, you'll have to disable the module first. For instance:

```
# update-alternatives --set php /usr/bin/php%version%
# update-alternatives --set phpize /usr/bin/phpize%version%
# update-alternatives --set php-config /usr/bin/php-config%version%
# phpdismod -v %version% imagick
# rm /etc/php/%version%/mods-available/imagick.ini
# pecl uninstall imagick
```

WARNING: In both cases, you must not forget to reset the various alternatives to their previous values, as required by i-MSCP core:

```
# PHP_VERSION=$(egrep '^PHP_VERSION' /etc/imscp/php/php.data | sed -e
's/PHP_VERSION\s\+=\s\+\(.*\)/\1/g')
# update-alternatives --set php /usr/bin/php"$PHP_VERSION"
# update-alternatives --set phpize /usr/bin/phpize"$PHP_VERSION"
# update-alternatives --set php-config /usr/bin/php-config"$PHP_VERSION"
```

IonCube loader

If you want to install this extension for one of your PHP version, you can process as follows (example for the x86-64 architecture):

For a compiled PHP version

```
# cd /usr/local/src
# wget
http://downloads3.ioncube.com/loader_downloads/ioncube_loaders_lin_x86-64.ta
r.gz
# tar -xzf ioncube_loaders_lin_x86-64.tar.gz
# cd ioncube
# PHP_EXT_DIR=$(/opt/phpswitcher/%ymd%/php%version%/bin/php-config --
extension-dir)
# cp ioncube_loader_lin_%version%.so $PHP_EXT_DIR/ioncube.so
# echo "zend_extension = $PHP_EXT_DIR/ioncube.so" >
/opt/phpswitcher/%ymd%/php%version%/etc/php/conf.d/01_ioncube.ini
```

Finally, you must restart the Web server or PHP-FPM instance, depending on the i-MSCP httpd server implementation in use.

For a packaged PHP version

```
# apt-get update
# apt-get install php%version%-dev
# cd /usr/local/src
# wget
http://downloads3.ioncube.com/loader_downloads/ioncube_loaders_lin_x86-64.ta
r.gz
# tar -xzf ioncube_loaders_lin_x86-64.tar.gz
# cd ioncube
# PHP_EXT_DIR=$(php-config%version% --extension-dir)
# cp ioncube_loader_lin_%version%.so "$PHP_EXT_DIR"/ioncube.so
# echo -e "; priority=01\nzend_extension = $PHP_EXT_DIR"/ioncube.so" >>
/etc/php/%version%/mods-available/ioncube.ini
# phpenmod -v %version% ioncube
```

Finally, you must restart the Web server or PHP-FPM instance, depending on the i-MSCP httpd server implementation in use.

PHP configuration information (phpinfo)

This feature allows the customers to access the configuration information of various PHP versions through their own PhpSwitcher interface. This feature can be disabled by editing the plugin configuration file, and by triggering a plugin list update through the plugin management interface.

With the i-MSCP FPM httpd server implementation

The plugin make use of the default www.conf FPM pool of each PHP version to serve the PHP configuration information at runtime.

With the i-MSCP Fcgid httpd server implementation

The plugin generate a static PHP info file for each PHP version. Because the information are static, you must not forget to re-generate them each time you do a configuration change for a specific PHP

version (e.g. when you enable or disable a PHP/PECL extension). This task can be done through the
PhpSwitcher administration interface, for each PHP version.

Registration of new PHP versions

You can register new PHP version either using the `php_compiler.pl` script (recommended), or
manually through the PhpSwitcher administration interface.

Registration through the `php_compiler.pl` script

To register a PHP version, you need to execute the following command:

```
# perl /var/www/imscp/gui/plugins/PhpSwitcher/PhpCompiler/php_compiler.pl --
register --install-dir %installdir%
```

See the previous documentation sections for all possible options, notably those which apply to
packaged PHP versions.

Registration through the PhpSwitcher administration interface

1. Login into the panel as administrator and go to the PhpSwitcher administration interface
   (settings section)

2. Setup your new PHP versions as follows:

   Common settings (Required)

   - Path to PEAR directory: /opt/phpswitcher/%ymd%/php%version%/share/pear

   PHP FastCGI settings (Required)

   - Path to binary: /opt/phpswitcher/%ymd%/php%version%/bin/php-cgi

   PHP-FPM settings (Required)

   - Path to binary: /opt/phpswitcher/%ymd%/php%version%/sbin/psw%version%-fpm
   - Path to configuration file: /opt/phpswitcher/%ymd%/php%version%/etc/php-fpm.conf
   - Path to pool directory: /opt/phpswitcher/%ymd%/php%version%/etc/php-fpm.d

Plugin translation

You can translate this plugin using a gettext translation editor such as `poedit`. Translation files are
located under the `./l10n` directory inside of this plugin archive. Once translated you can send us
your translation file (po file) for integration in future release.

Note that if no translation file exists for your localization in the `./l10n/po` directory, you must create
it first from the l10n/PhpSwitcher.pot file. Be aware that your file must be UTF-8, else, it won't be
accepted.

License

```
i-MSCP PhpSwitcher plugin
© 2014-2019 Laurent Declercq <l.declercq@nuxwin.com>
i-MSCP License <https://www.i-mscp.net/license-agreement.html>
```

From:
https://wiki.i-mscp.net/ - **i-MSCP Documentation**

Permanent link:
https://wiki.i-mscp.net/doku.php?id=plugins:phpswitcher&rev=1569668746

Last update: **2019/09/28 12:05**