

**A**masty

# Composer User Guide

This manual assumes that you have already installed and set up the Composer tool. If you haven't installed the Composer yet see the instruction on [How to install it](#).


To work with the Amasty Composer you need to create a **Customer Account** on Amasty.com (to get Access Keys).

---

## Extension Installation via Composer

**Step 1.** Log in to your server via a Command line:

```
work@machine-pc:~$ ssh yourserver.com
```

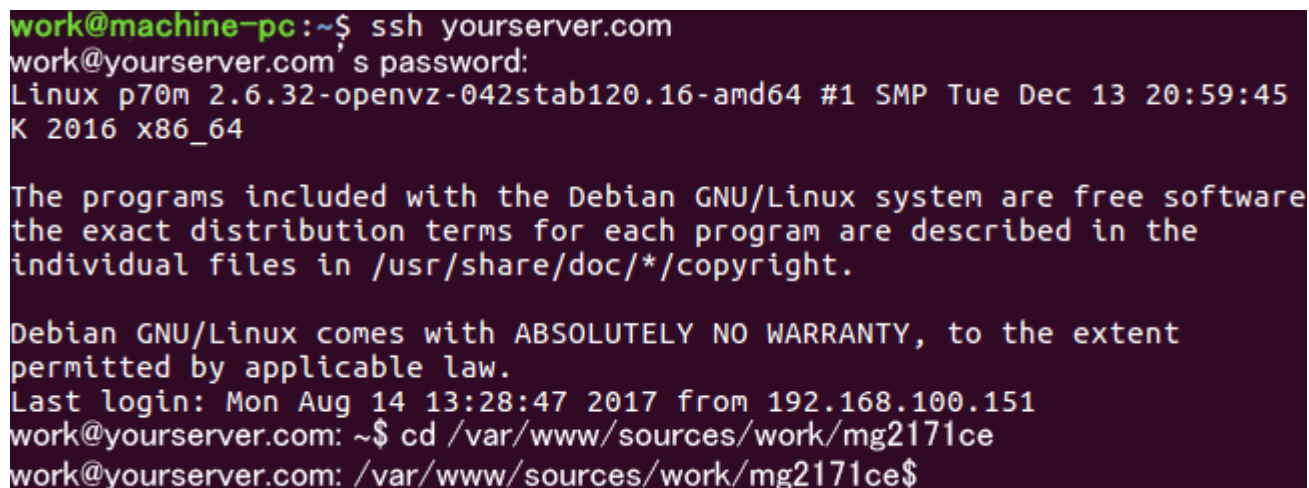


**Step 2.** Set the directory where you would like to install the extension. Usually, for such purposes, the Magento root folder is selected.

```
work@machine-pc:~$ ssh yourserver.com
work@yourserver.com's password:
Linux p70m 2.6.32-openvz-042stab120.16-amd64 #1 SMP Tue Dec 13 20:59:45
K 2016 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Aug 14 13:28:47 2017 from 192.168.100.151
work@yourserver.com: ~$ cd /var/www/sources/work/mg2171ce
work@yourserver.com: /var/www/sources/work/mg2171ce$
```



**Step 3.** Connect to the Amasty Composer Repository:

```
composer config repositories.amasty composer <path>
```

In the <path> indicate:

<https://composer.amasty.com/community/> - for community extensions.

<https://composer.amasty.com/enterprise/> - for enterprise extensions.

---

**Step 4.** To get authorized use Access Keys generated in your Customer Account :

**Amasty.com → Account → My Access Keys**

### MY ACCESS KEYS

[Create A New Access Key](#) These keys can be used for authorized downloads of Amasty Magento 2 extensions. Please see the [instruction document](#) for installing/updating

Product Name	Public Key	Private Key	Status	Actions
test	a649888fd42207f8acefda65505c5d29	f181e65a9f1d7e3695e1c9757f394933	Enabled	<a href="#">Disable</a> <a href="#">Delete</a>

Use the **Public Key** as a login and the **Private Key** - as a password.

You can save the keys in the repository, so you won't have to get authorized every time.

---

**Step 5.** After the authorization is successfully finished, you can start installing/updating your modules.

First, take a look at the module names in your **Customer Account** to make sure you call the necessary module via the **Composer** correctly.

Please go to **Amasty.com → Account → My Downloads** for paid extensions and open the **My free products** tab for free modules.

---

## MY DOWNLOADS

Extension	Platform	Latest Version		Technical help
<a href="#">Special Promotions Community</a>	Magento 2	1.2.7	<a href="#">Download ↓</a>	<a href="#">Extend support</a>
<a href="#">Special Promotions Pro Community</a>	Magento 1	1.11.0	<a href="#">Download ↓</a>	<a href="#">Extend support</a>

[Extend support for selected](#) [Alphabetical List View](#) [Group By Order](#)

## MY COMPOSER PACKAGES

Use Composer name when installing/updating Amasty Extensions for Magento 2 via Composer

Extension	Version	Composer Name
<a href="#">Special Promotions for Magento 2 Community</a>	1.2.7	amasty/module-special-promo

[← Back](#)

You will find module names for the Composer in the **'My Composer Packages'** section.

## MY FREE EXTENSIONS

Extension	Version Available to Download	
<a href="#">Product Alerts Reports</a>	1.0.2	<a href="#">Download ↓</a>
<a href="#">Promotions Manager</a>	1.0.2	<a href="#">Download ↓</a>
<a href="#">E-mail Blacklist</a>	1.0.1	<a href="#">Download ↓</a>
<a href="#">Easy Content Blocks</a>	1.0.10	<a href="#">Download ↓</a>
<a href="#">Product Categories</a>	1.0.0	<a href="#">Download ↓</a>
<a href="#">Search By Ranges and Categories</a>	1.0.0	<a href="#">Download ↓</a>
<a href="#">Detect Missing Pages</a>	2.0.12	<a href="#">Download ↓</a>
<a href="#">Facebook Reviews</a>	1.0.1	<a href="#">Download ↓</a>
<a href="#">Reviews Sidebar</a>	1.0.7	<a href="#">Download ↓</a>
<a href="#">Zero Product Sales</a>	1.0.2	<a href="#">Download ↓</a>
<a href="#">Color Swatches Pro for Magento 2</a>	1.1.3	<a href="#">Download ↓</a>
<a href="#">Google Invisible reCaptcha</a>	1.0.0	<a href="#">Download ↓</a>
<a href="#">Free Shipping After Discount</a>	1.0.1	<a href="#">Download ↓</a>
<a href="#">Facebook Comments for Magento 2</a>	1.0.0	<a href="#">Download ↓</a>
<a href="#">Google Invisible reCaptcha for Magento 2</a>	1.0.2	<a href="#">Download ↓</a>

## MY COMPOSER PACKAGES

Use Composer name when installing/updating Amasty Extensions for Magento 2 via Composer

Extension	Version	Composer Name
<a href="#">Color Swatches Pro for Magento 2</a>	1.1.3	amasty/color-swatches-pro
<a href="#">Facebook Comments for Magento 2</a>	1.0.0	amasty/fbpreview
<a href="#">Google Invisible reCaptcha for Magento 2</a>	1.0.2	amasty/aminvisiblecaptcha

**Step 6.** To install the module, please execute the following command:

```
composer require amasty/color-swatches-pro
```

```
work@yourserver.com: /var/www/sources/work/mg2171ce$ composer require amasty/
color-swatches-pro
Using version ^1.1 for amasty/color-swatches-pro
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
  - Installing amasty/color-swatches-pro (1.1.3): Downloading (100%)

Writing lock file
Generating autoload files
```

Instead of the '*color-swatches-pro*' please specify the module you need.

---

**Step 7.** To upgrade the module, please execute the command:

```
php bin/magento setup:upgrade
```

```
work@yourserver.com: /var/www/sources/work/mg2171ce$ php bin/magento setup:u
pgrade
Cache cleared successfully
File system cleanup:
/var/www/sources/work/mg2171ce/var/generation/Composer
/var/www/sources/work/mg2171ce/var/generation/Magento
/var/www/sources/work/mg2171ce/var/generation/Symfony
The directory '/var/www/sources/work/mg2171ce/var/di/' doesn't exist -
skipping cleanup
/var/www/sources/work/mg2171ce/pub/static/_requirejs
/var/www/sources/work/mg2171ce/pub/static/adminhtml
/var/www/sources/work/mg2171ce/pub/static/deployed_version.txt
/var/www/sources/work/mg2171ce/pub/static/frontend
/var/www/sources/work/mg2171ce/var/view_preprocessed/css
/var/www/sources/work/mg2171ce/var/view_preprocessed/source
Updating modules:
Schema creation/updates:
```

# Extension Update via Composer

The extension update process is the same as the module installation process. To update the extension, simply execute the command:

```
composer update <composer_name>
```

Instead of the '*composer\_name*' please specify the corresponding module name which is indicated in your customer account.

**IMPORTANT:** This command updates only the last two version numbers: e.g. 1.x.x ( only x.x will be updated).

To update the version from 1.x.x to 2.x.x you need to execute the following command:

```
composer require <composer_name> ^x..0 --update-with-dependencies
```

Where x corresponds to the first version number. For example, to update the Improved Layered Navigation Extension for M2 from the 1.x.x version to the 2.x.x version you need to execute the command:

```
composer require amasty/shopby ^2..0 --update-with-dependencies
```

To update all modules execute the command:

```
composer update
```

```
work@yourserver.com: /var/www/sources/work/mg2171ce$ composer update
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 0 installs, 21 updates, 0 removals
 - Updating zendframework/zend-stdlib (2.4.12 => 2.4.13): Loading from
cache
 - Updating zendframework/zend-eventmanager (2.4.12 => 2.4.13): Loading
from cache
 - Updating zendframework/zend-code (2.4.12 => 2.4.13): Loading from ca
che
 - Updating zendframework/zend-di (2.4.12 => 2.4.13): Loading from cach
e
 - Updating zendframework/zend-loader (2.4.12 => 2.4.13): Loading from
cache
 - Updating zendframework/zend-view (2.4.12 => 2.4.13): Loading from ca
che
 - Updating zendframework/zend-modulemanager (2.4.12 => 2.4.13): Loadin
g from cache
 - Updating zendframework/zend-uri (2.4.12 => 2.4.13): Loading from cac
he
 - Updating zendframework/zend-server (2.4.12 => 2.4.13): Loading from
cache
 - Updating zendframework/zend-soap (2.4.12 => 2.4.13): Loading from ca
che
 - Updating zendframework/zend-servicemanager (2.4.12 => 2.4.13): Loadi
ng from cache
 - Updating zendframework/zend-text (2.4.12 => 2.4.13): Loading from ca
che
 - Updating zendframework/zend-math (2.4.12 => 2.4.13): Loading from ca
che
 - Updating zendframework/zend-json (2.4.12 => 2.4.13): Loading from ca
che
 - Updating zendframework/zend-serializer (2.4.12 => 2.4.13): Loading f
rom cache
 - Updating zendframework/zend-log (2.4.12 => 2.4.13): Loading from cac
he
 - Updating zendframework/zend-i18n (2.4.12 => 2.4.13): Loading from ca
che
 - Updating zendframework/zend-console (2.4.12 => 2.4.13): Loading from
cache
 - Updating zendframework/zend-config (2.4.12 => 2.4.13): Loading from
cache
 - Updating magento/module-tax-sample-data (100.1.1 => 100.1.2): Downlo
ading (conDownloading (100%)
 - Updating zendframework/zend-filter (2.4.12 => 2.4.13): Loading from
cache
Writing lock file
Generating autoload files
```

# Delete Extension via Composer

To delete the extension via Composer, please execute the command:

```
composer delete <composer_name>
```

Instead of the '*composer\_name*' please specify the corresponding module name which is indicated in your customer account.