



Automatic Customer Group Switching Magento® 2 Extension

Installation Guide

Below you will find a step by step guide for the installation of the Magento® 2 Automatic Customer Group Switching extension purchased and downloaded from the Plugin Company website using the **command line**. Although this guide probably works with most third party plugins, we can't guarantee this as the instructions are specifically written for the Magento® 2 Automatic Customer Group Switching extension developed by Plugin Company.

Please follow the installation guide carefully to prevent errors in your production environment.

Always install the downloaded extension on a staging server first in order to detect any issues which might arise and make a backup of your production environment before migrating.

Table of Contents

Installing the Magento® 2 Extension

Installing the Magento® 2 Extension using Composer

Installing the Magento® 2 Extension

Download and Extract Installation Package

Log in to your Plugin Company account and download the extension's installation package to your local computer. We recommend to install the most recent extension version available.

Extract the installation package locally to any folder. The extracted archive should contain an **upload_contents** folder (the installation files), a license agreement and the extension's installation guide and user manual.

Upload Files to Server

Connect to your server using SSH, SFTP or FTP.

Copy the extracted **app folder** to your Magento® 2 root directory, using your preferred method of doing so.

Enable the Extension

First enable the license manager using the following command:

```
php bin/magento module:enable PluginCompany_LicenseManager
```

After enabling the license manager, enable the extension using the following command:

```
php bin/magento module:enable PluginCompany_CustomerGroupSwitching
```

Update the Database

Run the following upgrade command:

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

Clean the Magento® Cache

Clean the cache using the following command:

```
php bin/magento cache:clean
```

Remove auto-generated classes

Remove all auto-generated classes using the following command:

```
rm -Rf ./var/generation/*
```

Or in Magento® 2.2+

```
rm -Rf ./generated/*
```

Run the Compiler

Run the dependency injection compilation command:

```
php bin/magento setup:di:compile
```

Deploy Static View Files

Run the static content generation command:

```
php bin/magento setup:static-content:deploy
```

Installing the Magento® 2 Extension using Composer

Add the Plugin Company Composer Repository

Edit the `composer.json` file in the Magento® 2 root folder, and add the following to the `repositories` section:

```
{
  "type": "composer",
  "url": "https://plugin.company/"
}
```

The resulting section will look something like this after doing so:

```
"repositories": [
  {
    "type": "composer",
    "url": "https://plugin.company/"
  },
  {
    "type": "composer",
    "url": "https://repo.magento.com/"
  }
]
```

Important: Make sure the `plugin.company` repository is listed above the default `repo.magento.com` repository

If the `plugin.company` repository isn't listed first, composer will first check the Magento® repo for available packages and dependencies, resulting in the possible installation of outdated versions of these packages.

Install the Extension using Composer

Execute the following command to install the extension:

```
composer require plugincompany/customergroupswitching
```

When prompted for a username and password, fill in the credentials from your Plugin Company Repository Account.

Enable the Extension

First enable the license manager using the following command:

```
php bin/magento module:enable PluginCompany_LicenseManager
```

After enabling the license manager, enable the extension using the following command:

```
php bin/magento module:enable PluginCompany_CustomerGroupSwitching
```

Update the Database

Run the following upgrade command:

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

Clean the Magento® Cache

Clean the cache using the following command:

```
php bin/magento cache:clean
```

Remove auto-generated classes

Remove all auto-generated classes using the following command:

```
rm -Rf ./var/generation/*
```

Or in Magento® 2.2+

```
rm -Rf ./generated/*
```

Run the Compiler

Run the dependency injection compilation command:

```
php bin/magento setup:di:compile
```

Deploy Static View Files

Run the static content generation command:

```
php bin/magento setup:static-content:deploy
```

