

Catalog Permissions

Created by Peter Samoilov, last modified by Anton Lashuk on янв 05, 2019

- Installation
- Configuration
 - Global settings
 - Managing product and product price visibility depending on customer group
 - Managing categories visibility depending on customer group
- Market Segmentation Suite integration
- Uninstall
- Important notes
- Troubleshooting

Extension page: <http://ecommerce.aheadworks.com/magento-extensions/catalog-permissions.html>

Catalog Permissions is a great choice for those online merchants who want to configure the visibility of different items, product prices and even the whole categories for specified customer groups.

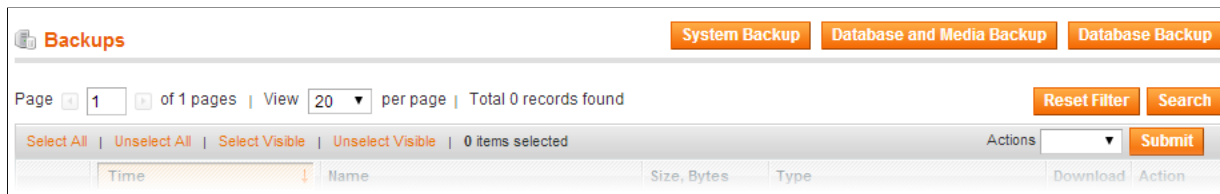
Installation

1. Backup your web directory and store database.

[Click to view details](#)

You can make backup copies with any tool you find appropriate

If you are going to use the native Magento backup function, navigate to **System -> Tools -> Backups** and perform **System** and **Database** backups



2. Log in to the Magento backend

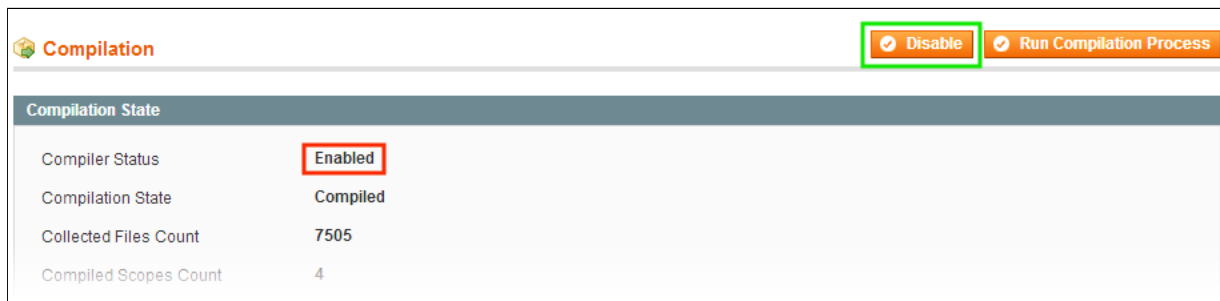
3. Disable compilation

[Click to view details](#)

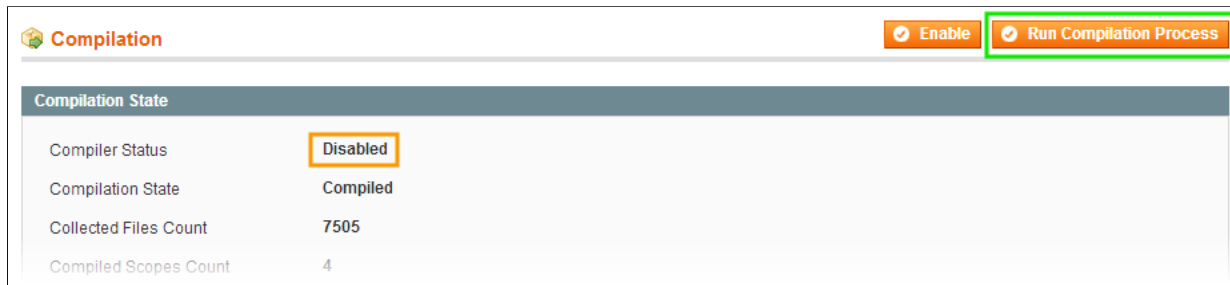
Navigate to **System -> Tools -> Compilation**.

If Compiler status is Disabled, you can skip to the next step

If Compiler is enabled, disable it.



IMPORTANT: after the extension is installed, you can enable the compilation again; **IT IS CRUCIAL** that you use "Run Compilation Process" function, not just "Enable button"



Installing an extension with the Compilation enabled will result in store downtime.

4. Flush store cache

[Click to view details](#)

You can flush the store cache in 2 ways:

- **via the backend:**

Navigate to **System -> Cache Management** menu, and click **Flush Magento Cache** button

Cache Storage Management

Flush Magento Cache
Flush Cache Storage

Select All | Unselect All | Select Visible | Unselect Visible | 0 items selected
Actions Refresh Submit

Cache Type	Description	Associated Tags	Status
<input type="checkbox"/> Configuration	System(config.xml, local.xml) and modules configuration files(config.xml).	CONFIG	ENABLED
<input type="checkbox"/> Layouts	Layout building instructions.	LAYOUT_GENERAL_CACHE_TAG	ENABLED
<input type="checkbox"/> Blocks HTML output	Page blocks HTML.	BLOCK_HTML	ENABLED

- **via filesystem:**

On your server, navigate to Magento root folder, then proceed to `/var/cache/`; delete all the content there.

5. Download the extension package from your account and extract the downloaded archive

6. Copy the content of `/Step_1/` folder to your store's root directory

If you are using a custom theme,
read the instructions here.

In case you are using a custom theme, it is recommended to copy the design files to your current theme's folders. In case there are several themes in use at the same store of them.

- Find this folder in the package: `/step_1/app/design/frontend/base/default/`; copy its content to `/app/design/frontend/[your_package]/[your_theme]/`
- Find this folder in the package: `/step_1/skin/frontend/base/default/`; copy its content to `/skin/frontend/[your_package]/[your_theme]/`

7. Copy the content of `/Step_2/` folder to your store's root directory

8. Flush store cache again; log out from the backend and log in again.

9. Set Up list.phtml

Click to view details

If you do not want products with hidden price to be displayed as **Out of stock** in catalog and search results lists, open the `/app/design/frontend/<your_package>/<your_theme>` having name of your current theme as `your_theme`, and your current package as `your_package`. If you haven't changed your theme or package, use `default` as the value.

Then find the following lines:

```
<p class="availability out-of-stock"><span><?php echo $this->__('Out of stock') ?></span></p>
```

and enclose them with a code like in the example below:

```
<?php if(!$product->getAWDisableOutOfStock()): ?>
  <p class="availability out-of-stock"><span><?php echo $this->__('Out of stock') ?></span></p>
<?php else: ?>
  <div class="aw-cat-perm-notify"><?php echo Mage::getStoreConfig('catalogpermissions/general/product_notice'); ?></div>
<?php endif; ?>
```

10. Wishlist view.phtml

For Magento Community
Magento CE > 1.4.x users

Open the following file: `/app/design/frontend/<your_package>/<your_theme>/template/wishlist/view.phtml`.

Magento CE > 1.7.x users

Open the following file: `/app/design/frontend/<your_package>/<your_theme>/template/wishlist/item/column/cart.phtml`

Then find the following lines:

```
<p class="availability out-of-stock"><span><?php echo $this->__('Out of stock') ?></span></p>
```

and enclose them with a code like in the example below:

```
<?php if(!$product->getAWDisableOutOfStock()): ?>
  <p class="availability out-of-stock"><span><?php echo $this->__('Out of stock') ?></span></p>
<?php else: ?>
  <div class="aw-cat-perm-notify"><?php echo Mage::getStoreConfig('catalogpermissions/general/product_notice'); ?></div>
<?php endif; ?>
```

For Magento Enterprise

Open the following file: `/app/design/frontend/base/default/template/wishlist/item/column/cart.phtml`.

Then find the following lines:

```
<?php if ($product->getIsSalable()): ?>
  <p class="availability in-stock"><span><?php echo $this->__('In stock') ?></span></p>
<?php else: ?>
  <p class="availability out-of-stock"><span><?php echo $this->__('Out of stock') ?></span></p>
```

```
<?php endif; ?>
```

And replace them with:

```
<?php if(!$product->getAWDisableOutOfStock()): ?>
<?php if ($product->getIsSalable()): ?>
    <p class="availability in-stock"><span><?php echo $this->__('In stock') ?></span></p>
<?php else: ?>
    <p class="availability out-of-stock"><span><?php echo $this->__('Out of stock') ?></span></p>
<?php endif; ?>
<?php else: ?>
    <div class="aw-cat-perm-notify"><?php echo Mage::getStoreConfig('catalogpermissions/general/product_notice'); ?></div>
<?php endif; ?>
```

For RWD theme

Open the following file: `/app/design/frontend/<your_package>/<your_theme>/template/wishlist/item/column/cart.phtml`.

Then find the following lines:

```
<?php if ($product->getIsSalable()): ?>
    <p class="availability in-stock">
        <span class="label"><?php echo $this->helper('catalog')->__('Availability:') ?></span>
        <span class="value"><?php echo $this->helper('catalog')->__('In stock') ?></span>
    </p>
<?php else: ?>
    <p class="availability out-of-stock">
        <span class="label"><?php echo $this->helper('catalog')->__('Availability:') ?></span>
        <span class="value"><?php echo $this->helper('catalog')->__('Out of stock') ?></span>
    </p>
<?php endif; ?>
```

And replace them with:

```
<?php if(!$product->getAWDisableOutOfStock()): ?>
<?php if ($product->getIsSalable()): ?>
    <p class="availability in-stock">
        <span class="label"><?php echo $this->helper('catalog')->__('Availability:') ?></span>
        <span class="value"><?php echo $this->helper('catalog')->__('In stock') ?></span>
    </p>
<?php else: ?>
    <p class="availability out-of-stock">
        <span class="label"><?php echo $this->helper('catalog')->__('Availability:') ?></span>
        <span class="value"><?php echo $this->helper('catalog')->__('Out of stock') ?></span>
    </p>
<?php endif; ?>
<?php else: ?>
    <div class="aw-cat-perm-notify"><?php echo Mage::getStoreConfig('catalogpermissions/general/product_notice'); ?></div>
<?php endif; ?>
```

11. Set Up View.phtml

For default theme

Open the following files:

`/app/design/frontend/<your_package>/<your_theme>/template/catalog/product/view.phtml` and IF EXISTS

`/app/design/frontend/<your_package>/<your_theme>/template/bundle/catalog/product/view.phtml`

Then find the following lines:

```
<?php echo $this->getChildHtml('product_type_data') ?>
```

and enclose them with a code like in the example below:

```
<?php if(!$product->getAWDisableOutOfStock()): ?>
    <?php echo $this->getChildHtml('product_type_data') ?>
<?php else: ?>
    <div class="aw-cat-perm-notify"><?php echo Mage::getStoreConfig('catalogpermissions/general/product_notice'); ?></div>
<?php endif; ?>
```

For RWD theme

Open the following files:

`/app/design/frontend/<your_package>/<your_theme>/template/catalog/product/view.phtml` and IF EXISTS

`/app/design/frontend/<your_package>/<your_theme>/template/bundle/catalog/product/view.phtml`

Then find the following lines:

```
<?php echo $this->getChildHtml('product_type_availability'); ?>
```

and replace them with:

```
<?php if(!$_product->getAWDisableOutOfStock()): ?>
  <?php echo $this->getChildHtml('product_type_availability'); ?>
<?php else: ?>
  <div class="aw-cat-perm-notify"><?php echo Mage::getStoreConfig('catalogpermissions/general/product_notice'); ?></div>
<?php endif; ?>
```

12. Set Up crosssell.phtml

For Magento Community

Open the following file: `/app/design/frontend/<your_package>/<your_theme>/template/checkout/cart/crosssell.phtml`

Then find the following lines:

```
<button type="button" title="<?php echo $this->__('Add to Cart') ?>" class="button btn-cart" onclick="setLocation('<?php echo $this->__($this->__('Add to Cart')) ?>')">
<span><span><?php echo $this->__('Add to Cart') ?></span></span></button>
```

and enclose them with a code like in the example below:

```
<?php if(!$_item->getAWDisableOutOfStock()): ?>
  <button type="button" title="<?php echo $this->__('Add to Cart') ?>" class="button btn-cart" onclick="setLocation('<?php echo $this->__($this->__('Add to Cart')) ?>')">
    <span><span><?php echo $this->__('Add to Cart') ?></span></span>
  </button>
<?php endif; ?>
```

For Magento Enterprise

Open the following file: `/app/design/frontend/<your_package>/<your_theme>/template/targetrule/checkout/cart/crosssell.phtml`

Then find the following lines:

```
<button type="button" title="<?php echo $this->__('Add to Cart') ?>" class="button btn-cart" onclick="setLocation('<?php echo $this->__($this->__('Add to Cart')) ?>')">
<span><span><?php echo $this->__('Add to Cart') ?></span></span></button>
```

And enclose them with a code like in the example below:

```
<?php if(!$_item->getAWDisableOutOfStock()): ?>
  <button type="button" title="<?php echo $this->__('Add to Cart') ?>" class="button btn-cart" onclick="setLocation('<?php echo $this->__($this->__('Add to Cart')) ?>')">
    <span><span><?php echo $this->__('Add to Cart') ?></span></span>
  </button>
<?php endif; ?>
```

13. Set Up Compare list.phtml

Click for details

Open the following file:

`/app/design/frontend/<your_package>/<your_theme>/template/catalog/product/compare/list.phtml`

Then find the following lines:

```
<p class="availability out-of-stock"><span><?php echo $this->__('Out of stock') ?></span></p>
```

and enclose them with a code like in the example below:

```
<?php if(!$_item->getAWDisableOutOfStock()): ?>
  <p class="availability out-of-stock"><span><?php echo $this->__('Out of stock') ?></span></p>
<?php else: ?>
  <div class="aw-cat-perm-notify"><?php echo Mage::getStoreConfig('catalogpermissions/general/product_notice'); ?></div>
<?php endif; ?>
```

Configuration

Global settings

Go to **System->Configuration->aheadWorks extensions->Catalog Permissions**

- **Hide price from group** - hides price for selected groups. Can be overridden with the similar option in a category or product settings (see description below).
- **Redirect from category page to** - here you can specify the page where customer with insufficient access will be redirected if they came by direct link.
- **Notice shown after redirection from category page** - here you can specify a message which customer will see. *Note, current version does not support native magento directives entered in this field, like `{{store direct_url=...}}` etc. However, HTML tags are partially supported.*
- **Redirect from product page to** - similar to the option **Redirect from category page to**
- **Notice shown after redirection from product page** - similar to **Notice to customer on redirect from category page**
- **Text displayed when price is hidden** - a text which will be displayed to customers with insufficient permissions. *"Log in to see the price"* by default.

Managing product and product price visibility depending on customer group

After the extension has been installed, store administrator can easily manage product and product price visibility for the specified customer groups in the Administration panel -> Catalog -> Manage Products -> Edit mode of any product -> Permissions tab.

Disable product for option hides product completely for the specified customer group

Hide product price for option hides only product price (as well as special price) and sets the product to the "Out of stock" status for the specified customer group

Managing categories visibility depending on customer group

After the extension has been installed, the **Permissions** tab appears at Administration panel -> Catalog -> Manage Categories
To set category visibility, set the **Hide category for a specific customer group** option to one or multiple values.

Market Segmentation Suite integration

Catalog Permissions extension can work in conjunction with Market Segmentation Suite(MSS) extension by aheadWorks.

If MSS installed new configuration fields that allow applying MSS rules come live at the category:

Click to see the screenshot

And product level Catalog Permissions settings:

Click to see the screenshot

Uninstall

1. **Backup your store database and web directory.**
2. Since Magento does not give ability to handle automatic uninstall for its modules, you need to run the following query on your database manually:

```
DELETE FROM `eav_attribute` WHERE `backend_model` = 'catalogpermissions/entity_attribute_backend_groups';
DELETE FROM `eav_attribute` WHERE `backend_model` = 'catalogpermissions/entity_attribute_backend_category';
DELETE FROM `eav_attribute` WHERE `backend_model` = 'catalogpermissions/entity_attribute_backend_segments';
```

3. This will delete extension elements from database.
4. Roll back the template changes described in chapter "Installation" (see above)
5. After that, you can safely remove the extension files from your store, starting from the file app/etc/modules/AW_Catalogpermissions.xml
6. Finally, run the following query in your database:

```
DELETE FROM `core_resource` WHERE `code` = 'catalogpermissions_setup';
```

This will delete the record that the extension has been installed on your Magento.

Important notes

If "Use Flat Catalog Category" or "Use Flat Catalog Product" options are enabled, reindex process should be launched for the application to work correctly.

Troubleshooting

After the extension installation the store gives an error, or blank page, or suggests to start Magento installation procedure.

Change the owner of the extracted extension files to the web server user and set 775 permissions on them. Clear the store cache and try again.

Right after installation I get 404 error in System->Configuration.

Logout from backend and login back again.

There is no aheadWorks extensions under my configuration section, or having the extension tab clicked I get a blank page, or Access Denied error.

Clear the store cache, browser cookies, logout and login again.

I've set up everything correctly, inserted the HTML code but there is nothing on that page.

Clear the store cache, clear your browser cache and domain cookies and refresh the page.

My configuration changes do not appear on the store.

Clear the store cache, clear your browser cache and domain cookies and refresh the page.

You can always find the latest version of the software, full documentation, demos, screenshots and reviews at <http://ecommerce.aheadworks.com>
License agreement: <http://ecommerce.aheadworks.com/LICENSE-M1.txt>
Contact us: <http://ecommerce.aheadworks.com/contacts/>
Copyright © 2019 aheadWorks Co. <http://www.aheadworks.com>

Нравится Станьте первыми кому понравится это

Ни одной