

InPost Integration

Magento installation guide

Created:	03/02/2018
Updated:	18/02/2019
Prepared By:	Sergii Khomenko
Version No:	1.5



1. Contents

1. Contents	2
2. Overview	3
3. Installation and configuration	3
3.1. Download module	3
3.2. Module installation	3
3.3. Default country	4
3.4. New order email confirmation	5
3.5. Success page	5
3.6. Initial APM import	5
4. Automated background processes	6
4.1. Nightly locations caching	6
4.2. Updating parcel statuses	7
5. Compatibility with 3rd party extensions	7
6. Fulfilment and order management	7
6.1. Standard Magento admin panel	7
6.2. 3 rd party order management systems	8
7. Logs	8
8. Contacts	9



2. Overview

InPost has a national network of nearly 1,200 fully automated parcel lockers that are accessible 24/7, meaning no more queues or waiting in, enabling customers to collect, send and return parcels at their earliest convenience. All lockers are located in a variety of safe and secure locations including Morrisons supermarkets, Esso petrol stations and Transport for London sites, as well as outside retailers such as Toys R Us.

This Magento module implements InPost Locker service allowing visitors to send their parcels to their locker of choice nearby.

With this extension, new shipping method is added where customer will be able to navigate lockers in their area and select the most convenient. Locker information is stored within customer order and is communicated to InPost via API during fulfillment and shipping process. Customers are able to track progress of their parcel from within 'My Account' area. Problematic parcels are reflected within admin system messages so your Customer Service can communicate and process these orders accordingly.

Demo environment can be seen at <http://www.magento-demo.inpost.nmediasystems.com/>

3. Installation and configuration

3.1. Download module

Latest release of the module can be downloaded from <https://github.com/inpostuk/magento/releases>

3.2. Module installation

1. Backup file system of your magento installation;
2. Copy app/ folder via ftp to your server into magento root folder on your development server. This will copy extension files to appropriate location. Please note no core files are affected.
3. Refresh system cache in System->Cache Management admin section;
4. Time to configure the extension. All configuration is done in the admin panel in Configuration section, System->Configuration->Shipping Methods->InPost Lockers.
 - a. Enabled;
 - b. Shipping price;
 - c. Title;
 - d. Shipping method name;
 - e. API Token – InPost account token, you should receive this token from InPost when account is created;



- f. Merchant email – InPost account email;
- g. API endpoint – either <https://stage-api-uk.easypack24.net/v4/> for testing purposes or <https://api-uk.easypack24.net/v4/> for production environment;
- h. Labels format, either PDF or ZPL if you use ZPL printer;
- i. Weight, this is weight dimension used in your store, either in kilograms or pounds;
- j. Default size class – depending on the size of items sold, you can select locker compartment size by default, A – 8x38x64cm, B – 19x38x64cm, C – 41x38x64cm;
- k. Sort order – positioning for InPost shipping method;
- l. Debug Mode – if debug mode enabled;
- m. Google Maps Api Key – as we're relying on Google Maps, you will need to provide your Google Maps Api Key, please register with [Google Maps](#);
- n. Default Map Radius – depending on the density of lockers in your customers target area, this setting will change default limit of lockers around entered post code;

Customers	Promotions	Newsletter	CMS	Reports	System
<p>but disabled. This may expose security risks. We strongly recommend to Enable</p> <p>haven't been picked up by customers from lockers yet (InPost Stored24)</p>					<p>Scheduler Admin / Security Section</p> <p>My Account have 4 critical, 6 major, 1!</p> <p>Notifications</p> <p>Tools</p> <p>Web Services</p> <p>Design</p> <p>Import/Export [WEBSITE]</p> <p>Manage Currency [STORE VIEW]</p> <p>Transactional Emails [STORE VIEW]</p> <p>Custom Variables [STORE VIEW]</p> <p>Permissions [STORE VIEW]</p> <p>Magento Connect [STORE VIEW]</p> <p>Cache Management [STORE VIEW]</p> <p>Index Management [STORE VIEW]</p> <p>Manage Stores [WEBSITE]</p> <p>Order Statuses [WEBSITE]</p> <p>Configuration [WEBSITE]</p>
<h3>Shipping Methods</h3>					
<h4>InPost Lockers</h4>					
Enabled	<input type="text" value="Yes"/>				
Shipping Price	<input type="text" value="2.50"/>				
Title	<input type="text" value="InPost 24/7 Lockers"/>				
Shipping Method Name	<input type="text" value="InPost Next Day Locker"/>				
API Token	<input type="text"/>				
Merchant email	<input type="text" value="demo@inpost.co.uk"/>				
API endpoint	<input type="text" value="https://stage-api-uk.eas"/>				
Labels Format	<input type="text" value="PDF"/>				
Weight	<input type="text" value="kg"/>				
Default size class	<input type="text" value="B"/> [WEBSITE]				
Sort Order	<input type="text" value="0"/> [STORE VIEW]				
Debug Mode	<input type="text" value="No"/> [WEBSITE]				
Google Maps Api Key	<input type="text"/> [WEBSITE]				
Default Map Radius	<input type="text" value="10"/> [STORE VIEW]				
▲ in miles					

3.3. Default country

When making address requests within map widget, we specify country which is taken from System->General->Countries Options->Default Country



Please make sure Default Country for your store is set to «United Kingdom» in order for the module to work properly within UK.

3.4. New order email confirmation

When new order is placed with InPost method, we'd like to notify customer about InPost specific collection, in order to display collection details, please add the following block to new order confirmation email template:

```
{code}
  {{block type="core/template" template="inpost/email/inpost.phtml" order=$order}}
{code}
```

3.5. Success page

Success page will also contain text about collection for InPost orders, we haven't overridden the template by default, please see new template with conditional InPost message included `app/design/frontend/base/default/template/inpost/checkout/success.phtml` between lines 40 and 42 and copy them across to your custom success page template.

3.6. Initial APM import

As described in the next section, list of InPost machines is imported / updated nightly. For the very first time lockers are imported automatically during extension installation. It's always possible to run this job manually from System->Scheduler->Job configuration, tick `inpost_locker_machines`, select «Schedule now» and hit «Submit» in the top right corner. The job will be scheduled and ran on the next cron execution which will import InPost machines and they'll be visible on the map widget.



Magento Admin Panel Global Record Search Logged in as admin | Friday, 16 March 2018 | [Log Out](#)

Dashboard Sales Catalog Customers Promotions Newsletter CMS Reports **System** [Get help for this page](#)

Important: Formkey validation on checkout disabled. This may expose security risks. We strongly recommend to Enable [Formkey validation](#) in the [Admin / Configuration / System](#) section, for protect your own checkout process.

Latest Message: InPost: 1 parcels with labels expired (InPost LabelExpired)

No default user is configured for who should run the cron. [Click here](#) to define one. We suggest using "www-d

Scheduler is working. (Last heart beat: 2 minute(s) ago)

Available Jobs

Select All | Unselect All | Select Visible | Unselect Visible | 1 items selected

Job code	Name	Short Description	Cron expression	Run mode	Parameters	Groups	Type	Status
<input type="checkbox"/> aggregate_sales_report_best sellers_data			0 0 ***	sales/o			xml	ENABLED
<input type="checkbox"/> aggregate_sales_report_coupons_data			0 0 ***	sales/o			xml	ENABLED
<input type="checkbox"/> aggregate_sales_report_invoiced_data			0 0 ***	sales/o			xml	ENABLED
<input type="checkbox"/> aggregate_sales_report_order_data			0 0 ***	sales/o			xml	ENABLED
<input type="checkbox"/> aggregate_sales_report_refunded_data			0 0 ***	sales/o			xml	ENABLED
<input type="checkbox"/> aggregate_sales_report_shipment_data			0 0 ***	sales/o			xml	ENABLED
<input type="checkbox"/> aggregate_sales_report_tax_data			0 0 ***	tax/obs			xml	ENABLED
<input type="checkbox"/> aoescheduler_heartbeat			* / 5 * * * *	aoe_scheduler/task_heartbeat:run		all	xml	ENABLED
<input type="checkbox"/> aoescheduler_testtask				aoe_scheduler/task_test:run			xml	ENABLED
<input type="checkbox"/> captcha_delete_expired_images			* / 10 * * * *	captcha/observer::deleteExpiredImages			xml	ENABLED
<input type="checkbox"/> captcha_delete_old_attempts			* / 30 * * * *	captcha/observer::deleteOldAttempts			xml	ENABLED
<input type="checkbox"/> catalog_product_alert			0 0 ***	productaler/observer::process			db_xml	ENABLED
<input type="checkbox"/> catalog_product_index_price_reindex_all			0 2 ***	catalog/observer::reindexProductPrices			xml	ENABLED
<input type="checkbox"/> catalogrule_apply_all			0 1 ***	catalogrule/observer::dailyCatalogUpdate			xml	ENABLED
<input type="checkbox"/> core_clean_cache			30 2 * * *	core/observer::cleanCache			xml	ENABLED
<input type="checkbox"/> core_email_queue_clean_up			0 0 ***	core/email_queue::cleanQueue			xml	ENABLED
<input type="checkbox"/> core_email_queue_send_all			* / 1 * * * *	core/email_queue::send			xml	ENABLED
<input type="checkbox"/> currency_rates_update			0 0 ***	directory/observer::scheduledUpdateCurrencyRates			db_xml	ENABLED
<input type="checkbox"/> customer_flowpassword			0 0 1 * *	customer/observer::deleteCustomerFlowPassword			xml	ENABLED
<input checked="" type="checkbox"/> inpost_lockers_machines			15 3 * * *	inpost_lockers/cron::getMachinesFromApi			xml	ENABLED
<input type="checkbox"/> inpost_lockers_update_order_statuses			15 8,12,16 * * *	inpost_lockers/cron::updateInpostOrderStatuses			xml	ENABLED

4. Automated background processes

There're 2 cronjobs added as part of this module, `inpost_lockers_machines` to cache locations and `inpost_lockers_update_order_statuses` to update order statuses.

First of all, Magento Cron has to be enabled, see more info in the official [documentation](#). We'd also recommend to install [AOE Scheduler](#) which visualizes cronjobs and makes their management a lot easier.

4.1. Nightly locations caching

In order to improve performance, there's cronjob `inpost_lockers_machines` which connects to InPost API on nightly basis at 3:15AM and downloads / updates details about APM locations. It's highly recommended to change default execution time from 3:15AM to some custom time at night so that all retailers using our extension don't connect to InPost



API at the same time. This is doable via admin within AOE Scheduler, System->Scheduler->Job Configuration->inpost_lockers_machines, scroll down to Cron expression.

The screenshot shows the InPost admin interface. At the top, there's a navigation bar with tabs: Customers, Promotions, Newsletter, CMS, Reports, and System (highlighted in orange). Below the navigation bar, a dropdown menu is open for the 'System' tab, listing various options: Scheduler, My Account, Notifications, Tools, Web Services, Design, Import/Export, Manage Currency, Transactional Emails, Custom Variables, Permissions, Magento Connect, Cache Management, Index Management, Manage Stores, Order Statuses, and Configuration. The 'Scheduler' option is selected, leading to a page titled 'Job "inpost_lockers_machines"'. This page has a 'General' section with fields for 'Job code *' (containing 'inpost_lockers_machines'), 'Name', and 'Short description'. Above this section, there are status messages: a yellow one stating 'No default user is configured for who should run the cron.' and a green one stating 'Scheduler is working. (Last heart beat: 3 minute(s) ago)'.

4.2. Updating parcel statuses

In order to keep retailer informed about parcels lifecycle, 3 times a day the module synchronises order statuses for all orders placed within the last 7 days (except orders in final statuses, InPost Delivered, etc). If there're orders with statuses requiring (customer services) attention, corresponding messages are added after each run to system messages within admin panel. Similarly to the previous step, it's advisable to amend standard execution times from 8:15,12:15,16:15 to similar times, different minutes to avoid loading InPost API at the same time.

5. Compatibility with 3rd party extensions

The shipping method is by default compatible with Magento standard onepage checkout. In addition the method is also compatible out-of-the-box with:

- OneStepCheckout from Idev, <https://www.onestepcheckout.com/>
 - In order to ensure compatibility, dependency needs to be included in app/etc/modules/Inpost_Lockers.xml:


```
<depends>Idev_OneStepCheckout /></depends>
```
- One Step Checkout from Amasty, <https://amasty.com/single-step-checkout.html>

6. Fulfilment and order management

6.1. Standard Magento admin panel



Once the module is installed, every shipment created in admin for orders with InPost shipping method will be communicated to InPost via API. Once shipment is created, you'll be able to print label in PDF or ZPL format depending on configuration.

Shipment submission:

Parcel

Weight kg

Size class **B**

Size A: 8 cm x 38 cm x 64 cm

Size B: 19 cm x 38 cm x 64 cm

Size C: 41 cm x 38 cm x 64 cm

Append Comments ☐

Email Copy of Shipment ☐

Submit Shipment

Label printing:

Shipping and Tracking Information

[Track this shipment](#)

InPost 24/7 Lockers - InPost Next Day Locker Collection Total Shipping Charges: £2.50

Print Shipping Label

Carrier	Title	Number	Status	Action
InPost 24/7 Lockers	Click and collect InPost	CP000001238UK	prepared	Delete

Custom Value

Add

6.2. 3rd party order management systems

If your orders are processed with 3rd party order management system and InPost is notified about new orders from 3rd party management system there will be no conflict. You can only utilize frontend functionality of this extension, capture locker information within orders as shipping address, this information will then be processed by your OMS.

In case where OMS is used, please disable cronjob which updates order statuses: `inpost_lockers_update_order_statuses`. This can be done via admin panel within AOE Scheduler, System->Scheduler->Job Configuration, and set "Disabled" status for `inpost_lockers_update_order_statuses` job.

7. Logs

All errors are logged to appropriate InPost log file in `var/log/inpost.log`



8. Contacts

If you have questions related to InPost account management or orders, please contact it_support@inpost.co.uk

For extension customization enquiries, please contact info@nmediasystems.com