

InPost Integration

Magento installation guide

Created on:	26/04/2018
Updated on:	17/04/2019
Prepared By:	Sergii Khomenko
Version No:	1.3.1



1. Contents

1. Contents	2
2. Overview	3
3. Installation and configuration	3
3.1. Module installation	3
3.2. Default country	5
3.3. New order email confirmation	5
3.4. Initial APM import	5
4. Automated background processes	5
4.1. Nightly locations caching	5
4.2. Updating parcel statuses	6
5. Fulfilment and order management	6
5.1. Standard Magento admin panel	6
5.2. 3 rd party order management systems	7
6. Logs	7
7. Contacts	7



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.

3. Installation and configuration

3.1. Module installation

1. Backup file system of your magento installation;
2. Installation
 - a. From zip archive:
 - i. marketplace.magento.com: Go to My profile / My purchases / InPost Lockers / Click Download
 - ii. Create directories Inpost/Lockers under app/code
 - iii. Copy extension's files into the Inpost/Lockers directory
 - iv. Run "composer require inpostuk/sdk-for-php:^1.0" from magento root directory
 - v. Run "composer update" from magento root directory
 - vi. Run "bin/magento setup:upgrade" from magento root directory.
 - b. Via composer:
 - i. in magento root directory run "composer require nmediasystems/module-inpostuk-lockers:^1.3 && composer update"
 - ii. Run "bin/magento setup:upgrade" from magento root directory.
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, Stores->Configuration->Sales->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;
- 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;

InPost Lockers



Enable [store view]	<input type="text" value="Yes"/>	<input type="checkbox"/> Use system value
Shipping Price [store view]	<input type="text" value="2.50"/>	<input type="checkbox"/> Use system value
Title [store view]	<input type="text" value="InPost 24/7 Lockers"/>	<input type="checkbox"/> Use system value
Shipping Method Name [store view]	<input type="text" value="InPost Next Day Locker Collection"/>	<input type="checkbox"/> Use system value
API Token [store view]	<input type="text"/>	<input type="checkbox"/> Use system value
Merchant Email [store view]	<input type="text"/>	<input type="checkbox"/> Use system value
API Endpoint [store view]	<input type="text" value="https://stage-api-uk.easypack24.net/v4/"/>	<input type="checkbox"/> Use system value
Labels Format [store view]	<input type="text" value="Pdf"/>	<input type="checkbox"/> Use system value
Weight [store view]	<input type="text" value="kg"/>	<input type="checkbox"/> Use system value
Size [store view]	<input type="text" value="A"/>	<input type="checkbox"/> Use system value
Sort Order [store view]	<input type="text" value="0"/>	<input type="checkbox"/> Use system value
Google Maps API Key [store view]	<input type="text"/>	<input type="checkbox"/> Use system value
Default Map API Radius [store view]	<input type="text" value="10"/>	<input type="checkbox"/> Use system value



3.2. 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.

3.3. 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 in Admin panel -> Marketing -> Communications -> Email templates. Add the following block to the new order confirmation email:

{code}

```
{{block class="Inpost\\Lockers\\Block\\Email\\Details  
Details" area='frontend' template="Inpost_Lockers::email/details.phtml" order=$order}}
```

{code}

3.4. Initial APM import

As described in the next section, list of InPost machines is imported / updated nightly. For the very first time the lockers are imported automatically during installation.

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 [Wyomind CronScheduler Pro](#) 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 can be changed in `app/code/inpost/Lockers/etc/crontab.xml` or cron job manager extension, such as Wyomind CronScheduler Pro.

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. This can be changed in `app/code/inpost/Lockers/etc/crontab.xml` or cron job manager extension, such as Wyomind CronScheduler Pro.

5. Fulfilment and order management

5.1. 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:

InPost parcel

Weight	Size class
<input type="text" value="2"/> kg	<input type="button" value="A"/>

Label printing:




Shipping and Tracking Information

[Track this shipment](#)

InPost 24/7 Lockers

Total Shipping Charges: **£2.50**

[Print Shipping Label](#)

Carrier	Title	Number	Action
InPost 24/7 Lockers	InPost 24/7 Lockers	CP000001324UK	
<input type="text" value="Custom Value"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Add"/>

5.2. 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 changed in `app/code/inpost/Lockers/etc/crontab.xml` or cron job manager extension, such as Wyomind CronScheduler Pro.

6. Logs

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

7. Contacts

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

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