Ingenico ePayments (Connect platform) for Magento® 1

This extension allows processing payments in Magento® 1 via the Ingenico ePayments Connect platform.

The document covers the **installation, configuration, and usage** of the extension.

End user documentation

1 Requirements

The following requirements must be met for the smooth operation of the module:

1.1 Magento® 1

The most recent versions of the following Magento® 1 versions are supported:

- Community Edition 1.8
- Community Edition 1.9

1.2 PHP

These PHP versions are supported:

- PHP 7.0
- PHP 5.6
- PHP 5.5

1.3 Ingenico contract

To use this extension, you need an Ingenico ePayments account (contract).

Please visit https://www.ingenico.com/epayments for more information or contact the Ingenico team.

2 Hints for using the module

2.1 Language support

The module supports the locale `en_US`.

2.2 Difference between Cancel and Void operations

On the order detail page, you will often have the ability to either "Cancel" or "Void" an order:

- To **cancel** an order means to cancel every transaction related to the order. This makes it impossible to create any further transactions (invoices, creditmemos, etc.).
- The **void** action only cancels the invoice of the order (i.e. a capture transaction). This leaves the order itself intact for further capture operations.

2.3 Compatibility with OneStepCheckout Magento® 1 Extension

The module is compatible with the OneStepCheckout Magento® 1 extension (onestepcheckout.com) in version 4.5.8. Other versions may work as well, but are not officially supported.

Before using the module with OneStepCheckout, the OneStepCheckout configuration must be set as follows:

- *AJAX update shipping / payment methods* is enabled,
- at least *Country* is selected in *AJAX save billing fields*,
- the options *Hide payment method* or *Exclude Country field* are **not** enabled.

Compatibility

We can only guarantee compatibility with the OneStepCheckout extension version 4.5.8. However, feel free to report any issues to our Technical support.

3 Installation and configuration

3.1 Before installation

We strongly recommend making a complete backup of the shop's files and database before any module installation or update.

Also, the installation and module functionality should be completely verified on a testing or staging system before moving it into a production system.

If you want to use Composer to install a Magento® 1 module, you'll need the Magento® 1 *Composer Installer* (https://github.com/Cotya/magento-composer-installer).

Please note

Basic questions about setting up and using Composer in Magento® 1 are not covered by our technical support. Please refer to the official documentation for Composer.

3.2 Installation options

Install the module's files according to your preferred setup/deployment strategy. It is recommended to use a version control system (e.g. Git).

See below for options and details on installing the module.

3.2.1 Installation from package using Composer

If you got the module as a `.zip` or `.tgz` package, and the package **contains a 'composer.json' file**, it is a so-called *Artifact* which can be installed with Composer.

Follow these steps:

- Copy (do not unzip) the package into a folder that is accessible by Composer.
- On the command line, navigate into the root folder of your Magento® 1 installation.
- Execute the following commands:

```
composer config repositories.epayments artifact /path/to/folder/with/zip/
composer require netresearch/connect-module-m1-ingco
composer update
```

- Make sure that no Composer errors occurred. If there are any, they must be resolved.
- **Important:** the path in the above command is absolute, i.e. starting at the server (Linux) root.
- Then log out from the Magento® 1 admin panel, and log in again.
- After this, clear all Magento® 1 caches.
- Now proceed with the Module configuration.

3.2.2 Installation from package, manually

If you have received the module as a `.zip` or `.tgz` package, and the package **does NOT contain a 'composer.json' file**, you can install it manually.

Follow these steps:

- Unpack the package contents into your Magento® 1 root directory.
- Verify that the module files have been copied into this directory:

```
app/code/community/Netresearch/Epayments
```

- Then log out from the Magento® 1 admin panel, and log in again.
- After this, clear all Magento® 1 caches.
- Now proceed with the Module configuration.

Please note

When installing the module this way, it is very important to use a version control system (e.g. Git). Otherwise, it will be more difficult to locate and remove the module files if you want to uninstall the module at a later time.

3.2.3 Installation from repository using Composer

The module is not yet available in a public repository.

Please follow the instructions for Installation from package, manually.

3.2.4 Installation using Modman

If you have the command line tool *modman* installed, you can use it to install the module.

Follow these steps:

- Extract the `.zip` or `.tgz` package into a directory that is accessible for modman.
- On the command line, navigate into the root folder of your Magento® 1 installation.
- Execute the following commands:

```
modman init
modman link /path/to/extracted/connect-module-m1-ingco
modman deploy connect-module-m1-ingco
```

- Then log out from the Magento® 1 admin panel, and log in again.
- After this, clear all Magento® 1 caches.
- Now proceed with the Module configuration.

Further info on modman installation without using a repository can be found here:
https://github.com/colinmollenhour/modman/wiki/Using-modman-without-git-or-svn

3.3 Module configuration

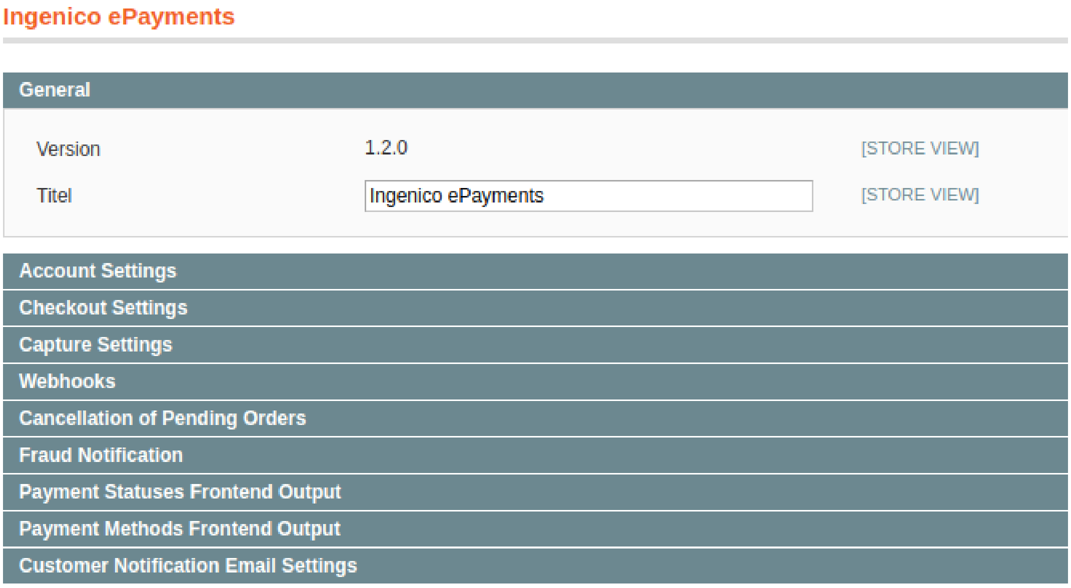
The extension adds a new configuration sections which must be configured to process payments with Ingenico ePayments:

```
System → Configuration → Sales → Ingenico ePayments
```

> **Please note**
>
> Make sure that all fields in *Account Settings* and *Fraud Notification* are configured before saving the configuration.
>
> Flush the Magento® 1 caches after making changes to the configuration.

3.3.1 General Settings

In the configuration section *General Settings*, you can configure the name of the Ingenico payment method that will be displayed to customers during checkout (see also Order placement).



Ingenico ePayments

General

Version 1.2.0 [STORE VIEW]

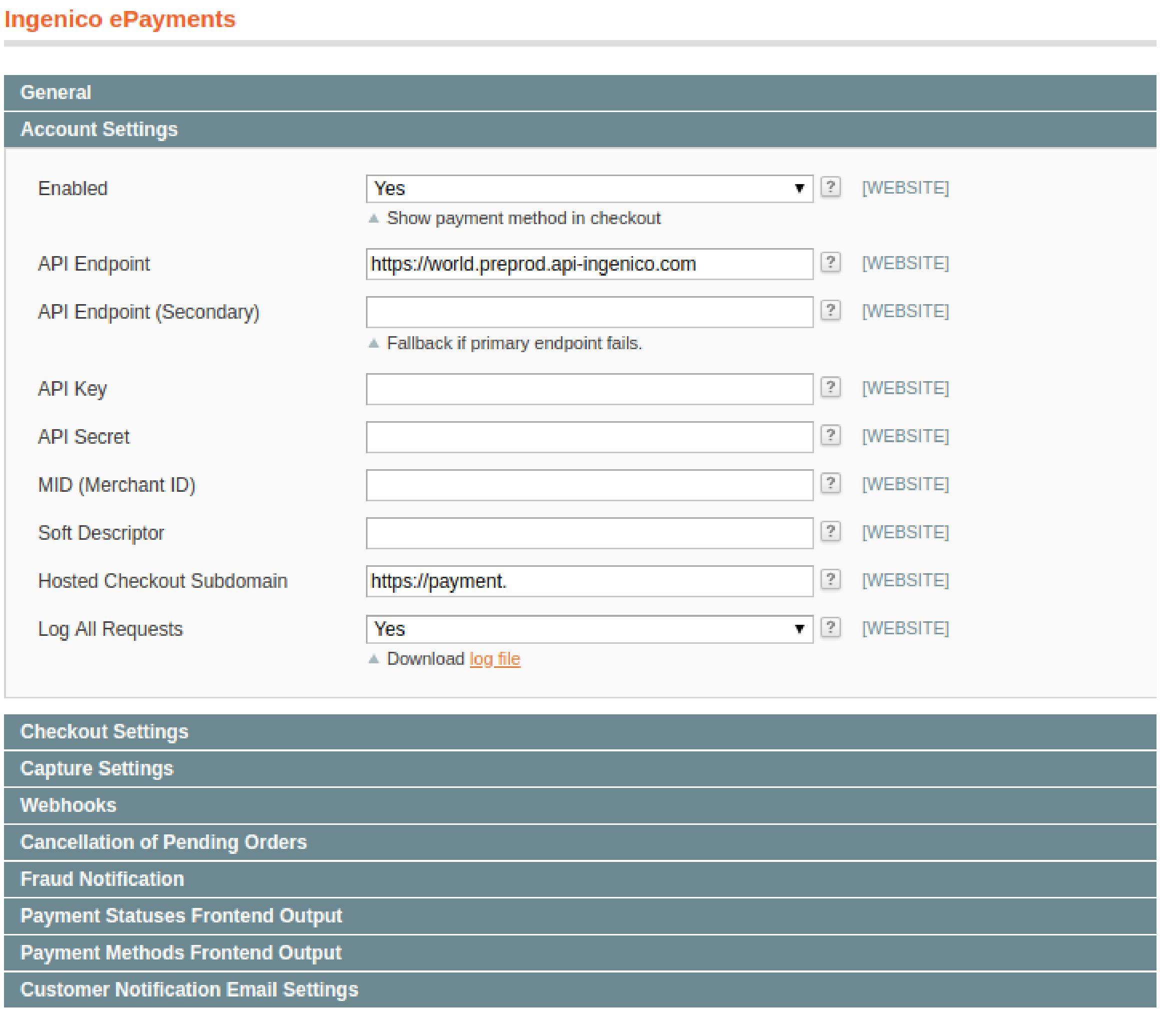
Titel Ingenico ePayments [STORE VIEW]

Account Settings
Checkout Settings
Capture Settings
Webhooks
Cancellation of Pending Orders
Fraud Notification
Payment Statuses Frontend Output
Payment Methods Frontend Output
Customer Notification Email Settings

3.3.2 Account Settings

The section *Account Settings* holds your access data (API endpoint, credentials, etc.) for the Ingenico ePayments platform. You will receive this information from the Ingenico team.

- Set *Enabled* to *Yes* to display Ingenico ePayments in the checkout.
- If you set *No*, the payment method is hidden from the checkout, but the module is still active.
- Fill the fields with the data you received from the Ingenico team.



Ingenico ePayments

General

Account Settings

Enabled	Yes ▾ ▲ Show payment method in checkout	[WEBSITE]
API Endpoint	https://world.preprod.api-ingenico.com	[WEBSITE]
API Endpoint (Secondary)	 ▲ Fallback if primary endpoint fails.	[WEBSITE]
API Key		[WEBSITE]
API Secret		[WEBSITE]
MID (Merchant ID)		[WEBSITE]
Soft Descriptor		[WEBSITE]
Hosted Checkout Subdomain	https://payment.	[WEBSITE]
Log All Requests	Yes ▾ ▲ Download log file	[WEBSITE]

Checkout Settings

Capture Settings

Webhooks

Cancellation of Pending Orders

Fraud Notification

Payment Statuses Frontend Output

Payment Methods Frontend Output

Customer Notification Email Settings

The secondary API endpoint is used as fallback if authentication with the primary endpoint fails. This allows to safely migrate endpoints.

When saving the configuration, these settings will automatically be validated (tested) against the API.

The value for Hosted Checkout Subdomain should usually remain as it is, ending with a period.

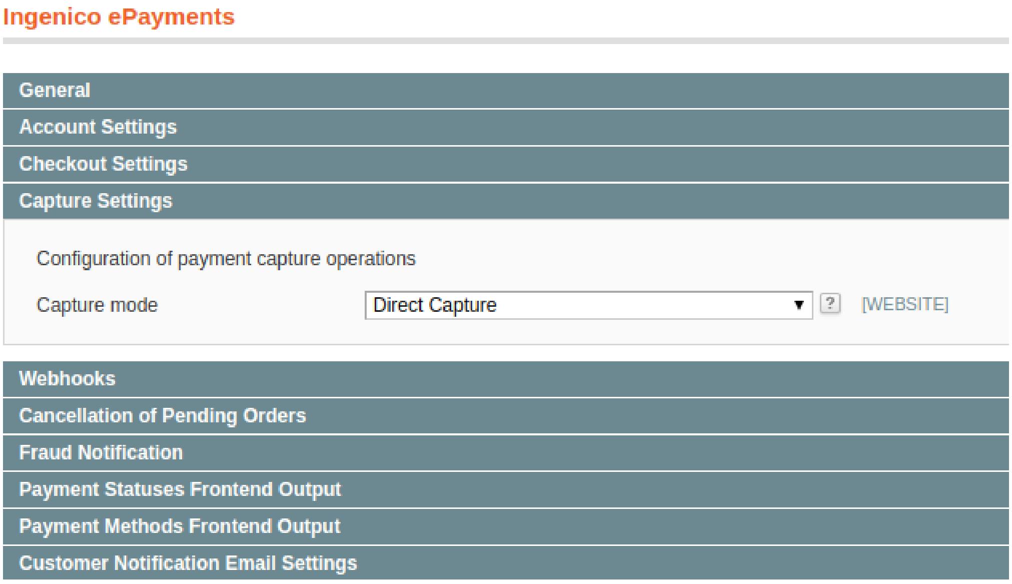
The System Identifier Prefix can be used to distinguish between several shop system instances. It applies system-wide to the entire Magento® 1 shop, not just to an individual Store or StoreView. The extension transmits Magento's order increment id as *merchantReference* to the platform. If a prefix is configured here, it is added before the order increment id, and the whole string is sent to the platform.

Please note

If you enable *Log All Requests*, you should clear or rotate the log files regularly.

3.3.3 Capture Settings

The section *Capture Settings* is used to configure what to do when a new payment is created. Here you have two options:



Ingenico ePayments

General
Account Settings
Checkout Settings
Capture Settings

Configuration of payment capture operations

Capture mode: Direct Capture [WEBSITE]

Webhooks
Cancellation of Pending Orders
Fraud Notification
Payment Statuses Frontend Output
Payment Methods Frontend Output
Customer Notification Email Settings

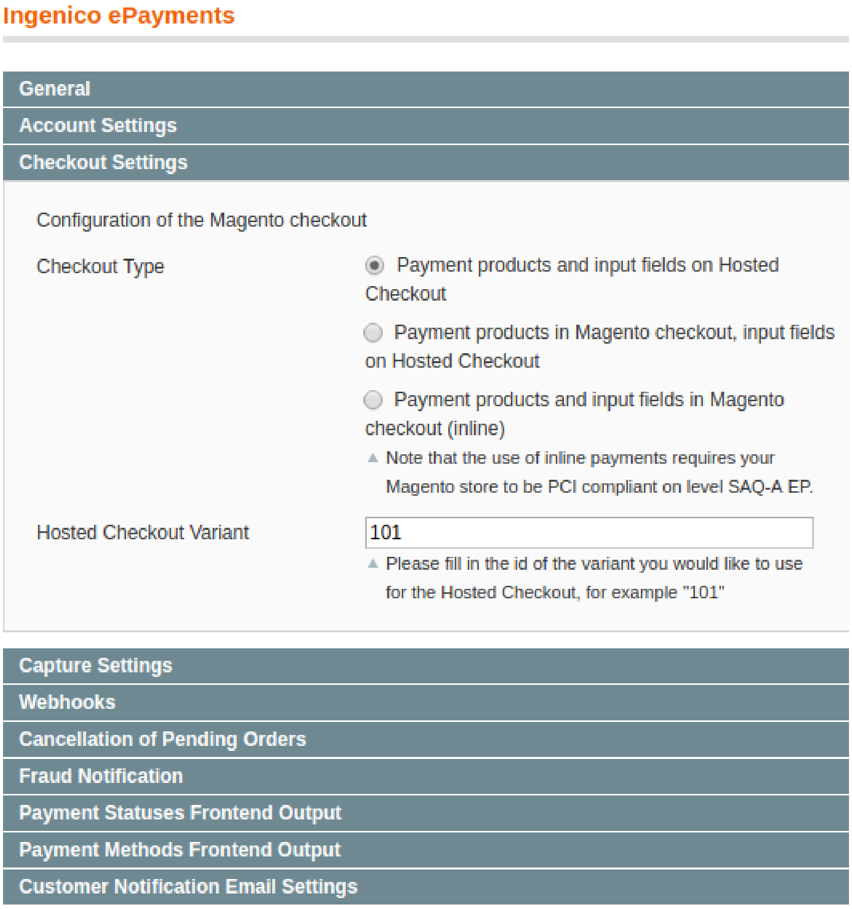
- *Direct Capture*: payments will automatically be captured and will be reported as paid as soon as the capture is complete on the gateway. An invoice is created automatically.
- *Delayed Settlement*: you will have to manually capture new payments by creating an invoice from the order details page.

Delayed settlements

For more information about delayed settlement, please refer to the Ingenico ePayments API Documentation.

3.3.4 Checkout Settings

The section *Checkout Settings* lets you configure how the payment products are displayed and where the payment data is going to be entered by the customer.



Ingenico ePayments

Configuration of the Magento checkout

Checkout Type
● Payment products and input fields on Hosted Checkout
○ Payment products in Magento checkout, input fields on Hosted Checkout
○ Payment products and input fields in Magento checkout (inline)
▲ Note that the use of inline payments requires your Magento store to be PCI compliant on level SAQ-A EP.

Hosted Checkout Variant: 101
▲ Please fill in the id of the variant you would like to use for the Hosted Checkout, for example "101"

Capture Settings
Webhooks
Cancellation of Pending Orders
Fraud Notification
Payment Statuses Frontend Output
Payment Methods Frontend Output
Customer Notification Email Settings

- *Payment products and input fields on Hosted Checkout:* Display available payment products with input fields on the Hosted Checkout page (external payment site) after placing the order.
- *Payment products in |mage| checkout, input fields on Hosted Checkout:* Display payment products (except input fields) directly in the Magento® 2 checkout. The customer enters the payment data afterwards on the Hosted Checkout page after placing the order.
- *Payment products and input fields in |mage| checkout (inline):* Display payment products with input fields directly in the Magento® 2 checkout (no redirection to external payment page, except for 3D Secure etc.).

The Hosted Checkout Variant is the numeric ID of the variant (template) to be used for the Hosted Checkout, e.g. "101". If none is configured, the default template that is set on the Connect account configuration will be used automatically.

3.3.4.1 Technical background

If you enable inline payments the payment data (e.g. credit card information) is requested directly in your checkout and encrypted via the Connect JavaScript SDK. The encrypted data is then sent to your shop and forwarded to the Connect platform through the Connect Server SDK. Not all payment methods support payment data capture through this way - for those who don't the Hosted Checkout page is being used.

3.3.4.2 PCI compliance

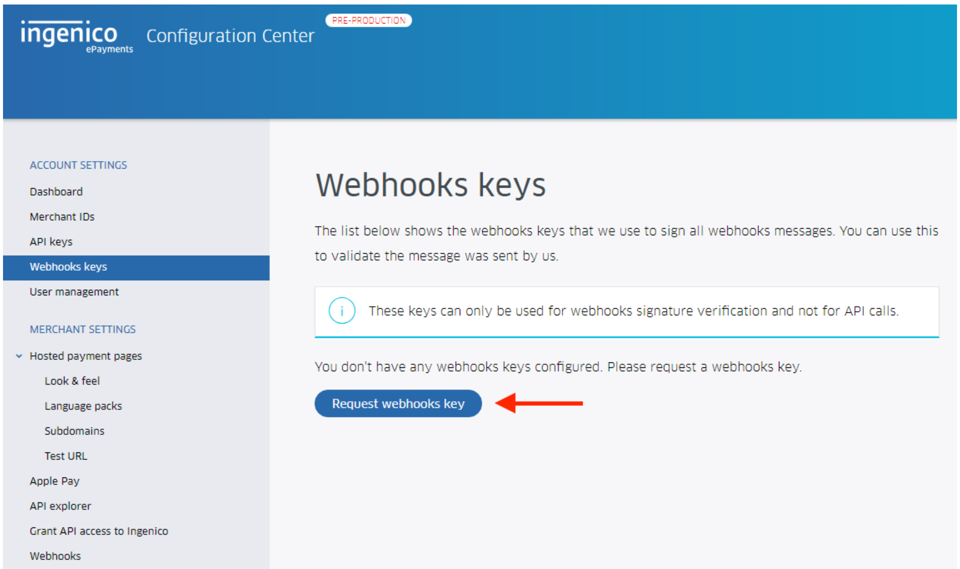
When using inline payments you will need to fill PCI SAQ A-EP, otherwise you should use PCI SAQ A.

3.3.5 Webhooks

In the section *Webhooks* you can set up the Ingenico Webhooks integration. If required, you will receive Webhooks credentials from the Ingenico ePayments team.

3.3.5.1 Webhook configuration at Ingenico

Please login to the Configuration Center and click on *Webhooks keys* on the left menu under *ACCOUNT SETTINGS*. Then click the button *Request webhooks key*.



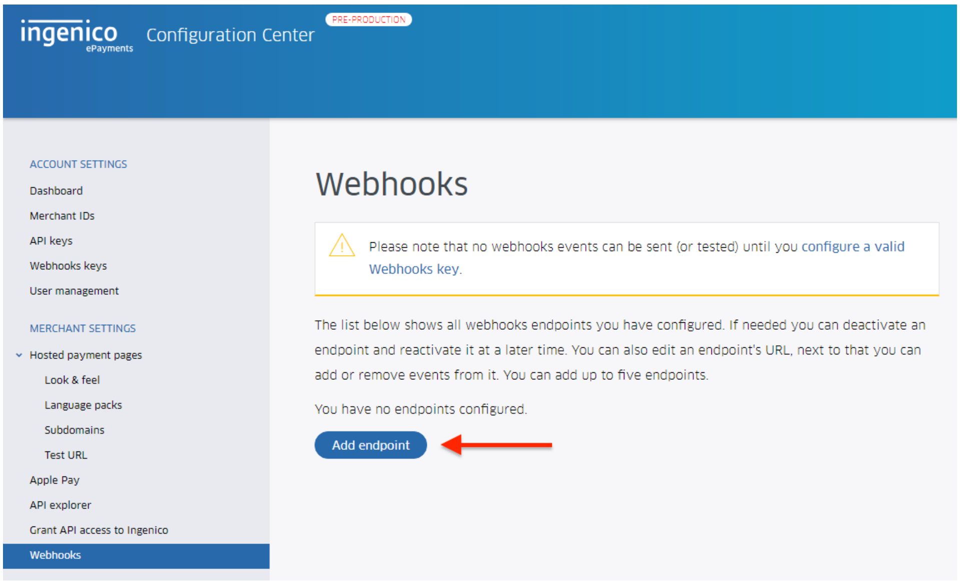
Webhooks keys

The list below shows the webhooks keys that we use to sign all webhooks messages. You can use this to validate the message was sent by us.

These keys can only be used for webhooks signature verification and not for API calls.

You don't have any webhooks keys configured. Please request a webhooks key.

To configure the endpoint, click on *Webhooks* on the left menu under *MERCHANT SETTINGS*. Then click the button *Add endpoint*.



Webhooks

Please note that no webhooks events can be sent (or tested) until you configure a valid Webhooks key.

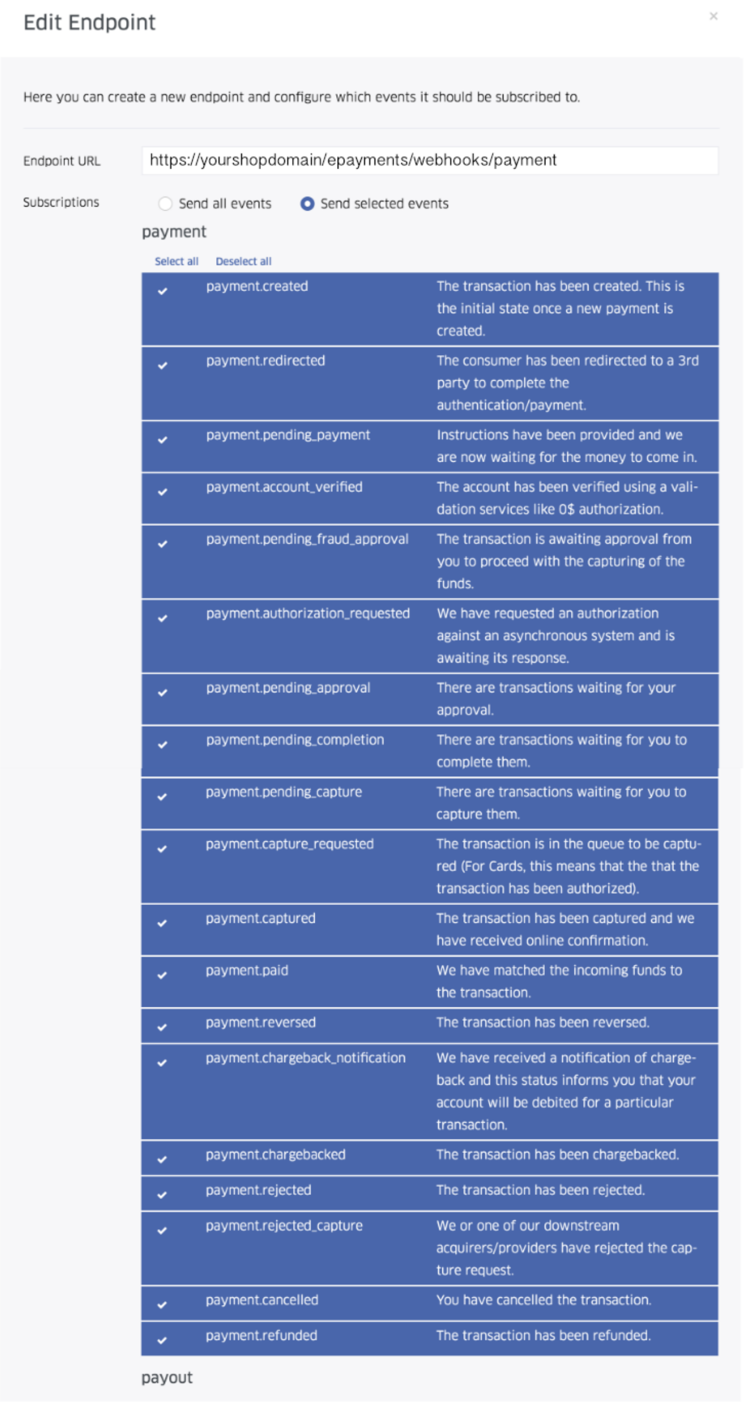
The list below shows all webhooks endpoints you have configured. If needed you can deactivate an endpoint and reactivate it at a later time. You can also edit an endpoint's URL, next to that you can add or remove events from it. You can add up to five endpoints.

You have no endpoints configured.

You will need to add two endpoints, one for the *payments* and one for the *refunds*.

3.3.5.2 Endpoint for payment

When you create the endpoint for *payment*, enter https://shop.domain/epayments/webhooks/payment (replace shop.domain with the correct domain) as the Endpoint URL and subscribe to the payment events:



Edit Endpoint ×

Here you can create a new endpoint and configure which events it should be subscribed to.

Endpoint URL: https://yourshopdomain/epayments/webhooks/payment

Subscriptions: ○ Send all events ● Send selected events

payment

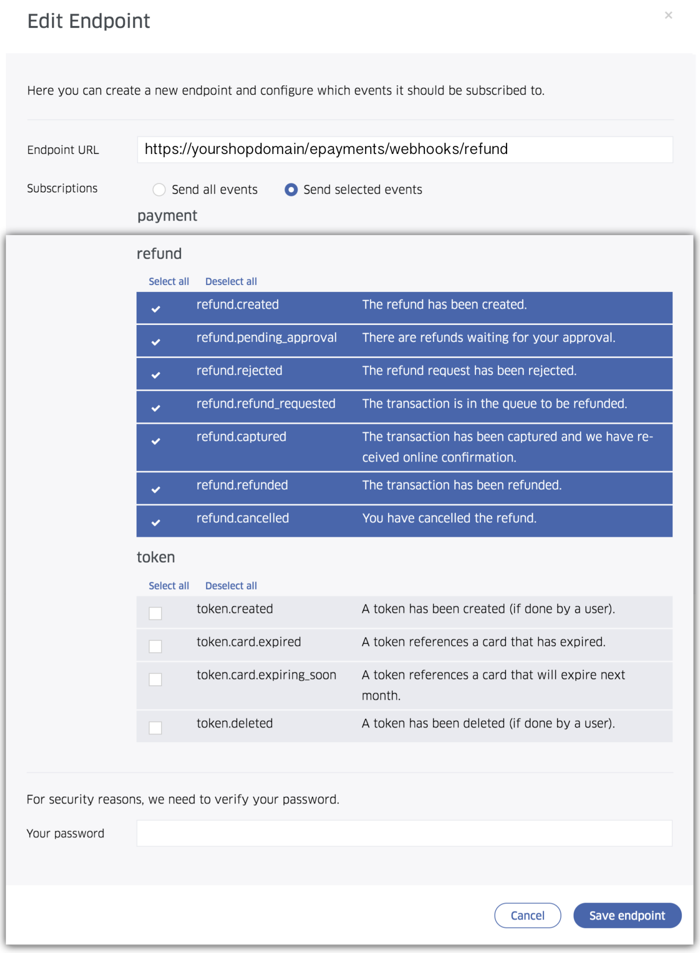
Select all Deselect all

✓	payment.created	The transaction has been created. This is the initial state once a new payment is created.
✓	payment.redirected	The consumer has been redirected to a 3rd party to complete the authentication/payment.
✓	payment.pending_payment	Instructions have been provided and we are now waiting for the money to come in.
✓	payment.account_verified	The account has been verified using a validation services like 0$ authorization.
✓	payment.pending_fraud_approval	The transaction is awaiting approval from you to proceed with the capturing of the funds.
✓	payment.authorization_requested	We have requested an authorization against an asynchronous system and is awaiting its response.
✓	payment.pending_approval	There are transactions waiting for your approval.
✓	payment.pending_completion	There are transactions waiting for you to complete them.
✓	payment.pending_capture	There are transactions waiting for you to capture them.
✓	payment.capture_requested	The transaction is in the queue to be captured (For Cards, this means that the that the transaction has been authorized).
✓	payment.captured	The transaction has been captured and we have received online confirmation.
✓	payment.paid	We have matched the incoming funds to the transaction.
✓	payment.reversed	The transaction has been reversed.
✓	payment.chargeback_notification	We have received a notification of chargeback and this status informs you that your account will be debited for a particular transaction.
✓	payment.chargebacked	The transaction has been chargebacked.
✓	payment.rejected	The transaction has been rejected.
✓	payment.rejected_capture	We or one of our downstream acquirers/providers have rejected the capture request.
✓	payment.cancelled	You have cancelled the transaction.
✓	payment.refunded	The transaction has been refunded.

payout

3.3.5.3 Endpoint for refund

Create another endpoint for *refund*. Enter https://shop.domain/epayments/webhooks/refund (replace shop.domain with the correct domain) as the Endpoint URL and subscribe to the refund events.



Edit Endpoint

Here you can create a new endpoint and configure which events it should be subscribed to.

Endpoint URL: https://yourshopdomain/epayments/webhooks/refund

Subscriptions: ○ Send all events ● Send selected events

payment

refund

☑	refund.created	The refund has been created.
☑	refund.pending_approval	There are refunds waiting for your approval.
☑	refund.rejected	The refund request has been rejected.
☑	refund.refund_requested	The transaction is in the queue to be refunded.
☑	refund.captured	The transaction has been captured and we have received online confirmation.
☑	refund.refunded	The transaction has been refunded.
☑	refund.cancelled	You have cancelled the refund.

token

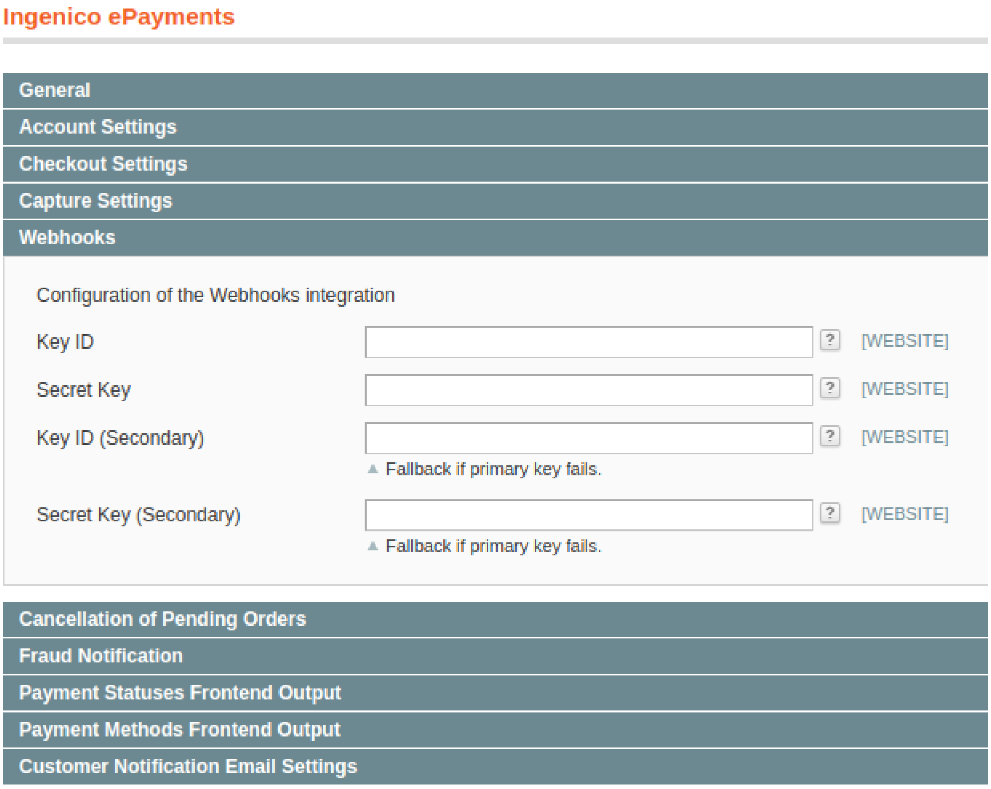
□	token.created	A token has been created (if done by a user).
□	token.card.expired	A token references a card that has expired.
□	token.card.expiring_soon	A token references a card that will expire next month.
□	token.deleted	A token has been deleted (if done by a user).

For security reasons, we need to verify your password.

Your password:

3.3.5.4 Webhook configuration in Magento® 1

In the Magento® 1 extension, you will have to enter the Webhooks Key ID and Secret Key in the following screen:



Ingenico ePayments

General

Account Settings

Checkout Settings

Capture Settings

Webhooks

Configuration of the Webhooks integration

Key ID ? [WEBSITE]

Secret Key ? [WEBSITE]

Key ID (Secondary) ? [WEBSITE]

▲ Fallback if primary key fails.

Secret Key (Secondary) ? [WEBSITE]

▲ Fallback if primary key fails.

Cancellation of Pending Orders

Fraud Notification

Payment Statuses Frontend Output

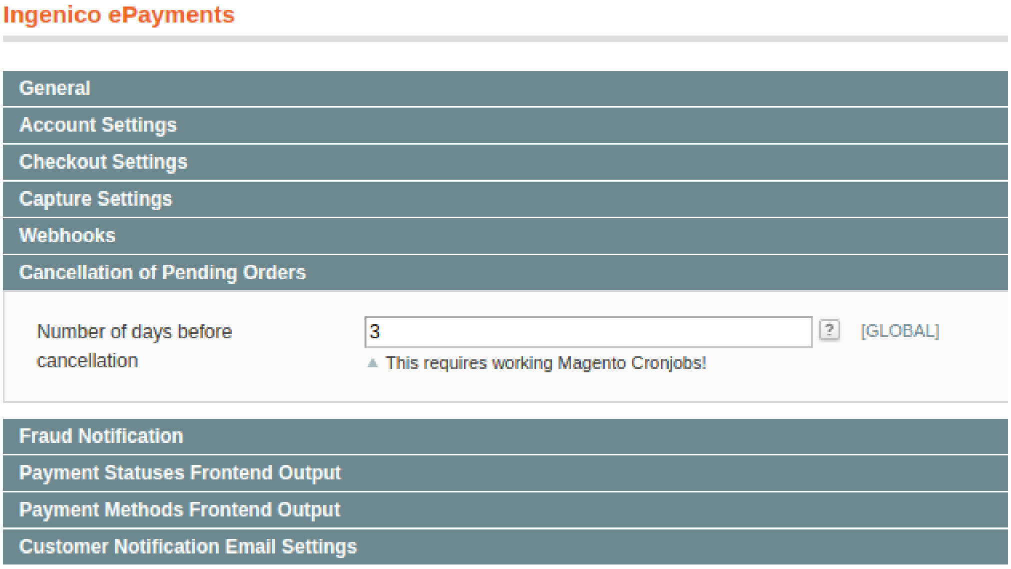
Payment Methods Frontend Output

Customer Notification Email Settings

Again, the *secondary* credentials are used if authentication with the *primary* credentials fails. You can use this to safely migrate keys without missing any Webhooks requests.

3.3.6 Cancellation of Pending Orders

In the section *Cancellation of Pending Orders*, you can configure the extension to automatically cancel abandoned orders after a period of days.



Ingenico ePayments

General
Account Settings
Checkout Settings
Capture Settings
Webhooks
Cancellation of Pending Orders

Number of days before cancellation: 3 [GLOBAL]

▲ This requires working Magento Cronjobs!

Fraud Notification
Payment Statuses Frontend Output
Payment Methods Frontend Output
Customer Notification Email Settings

Please note

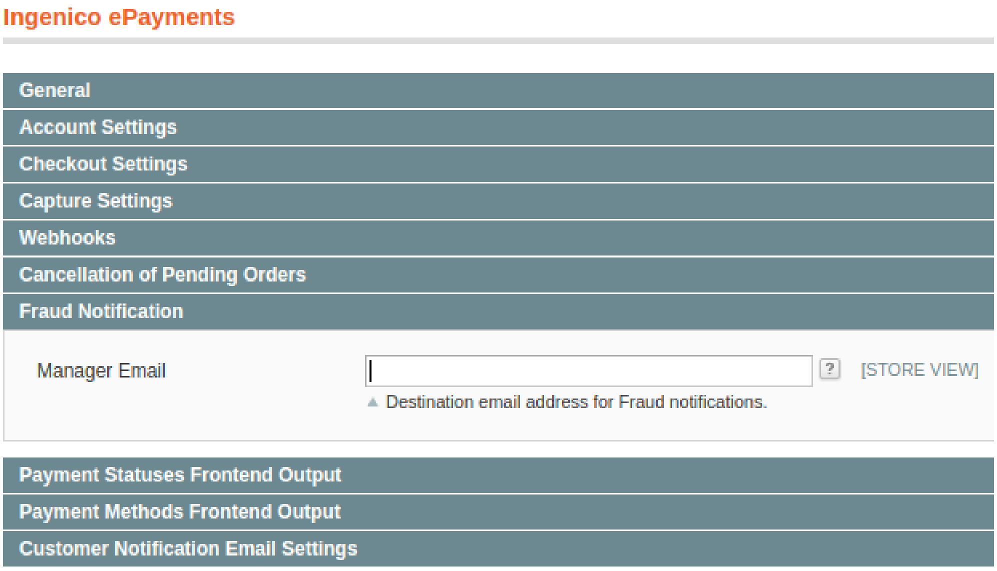
Automatic cancellations are performed by a periodically running *Cronjob*.

For this, the standard Magento® 1 Cronjobs must be configured correctly on your system. See also section Order update.

Please note that questions about the basic setup of Cronjobs in Magento® 1 are not covered by our technical support.

3.3.7 Fraud notification

In the section *Fraud notification* you can configure a destination email address for notifications when the Ingenico ePayments Fraud Detection is triggered.



Ingenico ePayments

General

Account Settings

Checkout Settings

Capture Settings

Webhooks

Cancellation of Pending Orders

Fraud Notification

Manager Email [STORE VIEW]

▲ Destination email address for Fraud notifications.

Payment Statuses Frontend Output

Payment Methods Frontend Output

Customer Notification Email Settings

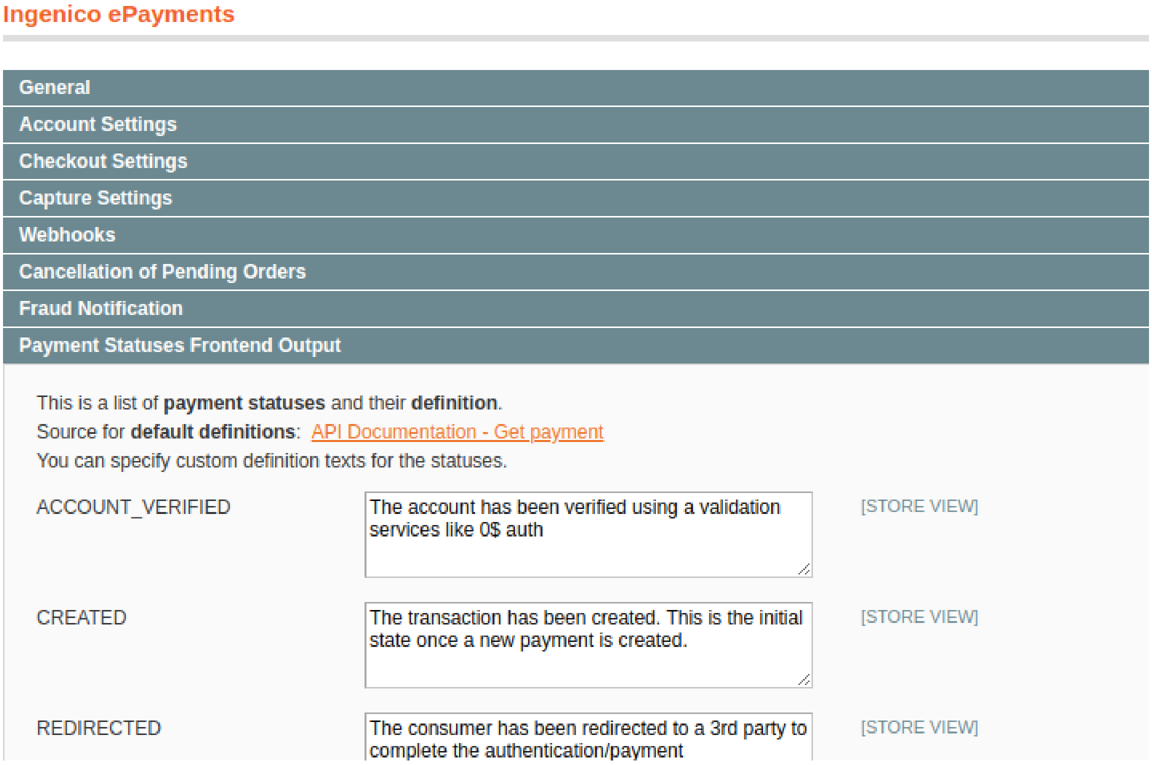
Please note

Manager Email is a required field.

If Magento® 1 doesn't allow saving the configuration, check if the field contains an email address.

3.3.8 Payment statuses frontend output

In the section *Payment statuses* you will find a list of payment statuses and their definitions.



Ingenico ePayments

This is a list of **payment statuses** and their **definition**.
Source for **default definitions**: API Documentation - Get payment
You can specify custom definition texts for the statuses.

ACCOUNT_VERIFIED	The account has been verified using a validation services like 0$ auth	[STORE VIEW]
CREATED	The transaction has been created. This is the initial state once a new payment is created.	[STORE VIEW]
REDIRECTED	The consumer has been redirected to a 3rd party to complete the authentication/payment	[STORE VIEW]

These definition texts will be displayed to the customer after making a payment in the Magento® 1 checkout.

You can modify the status definition texts on the Store View level according to your needs.

The default definitions are provided by Ingenico ePayments. If you need to know the original values, you can find them in the Ingenico ePayments API Documentation:

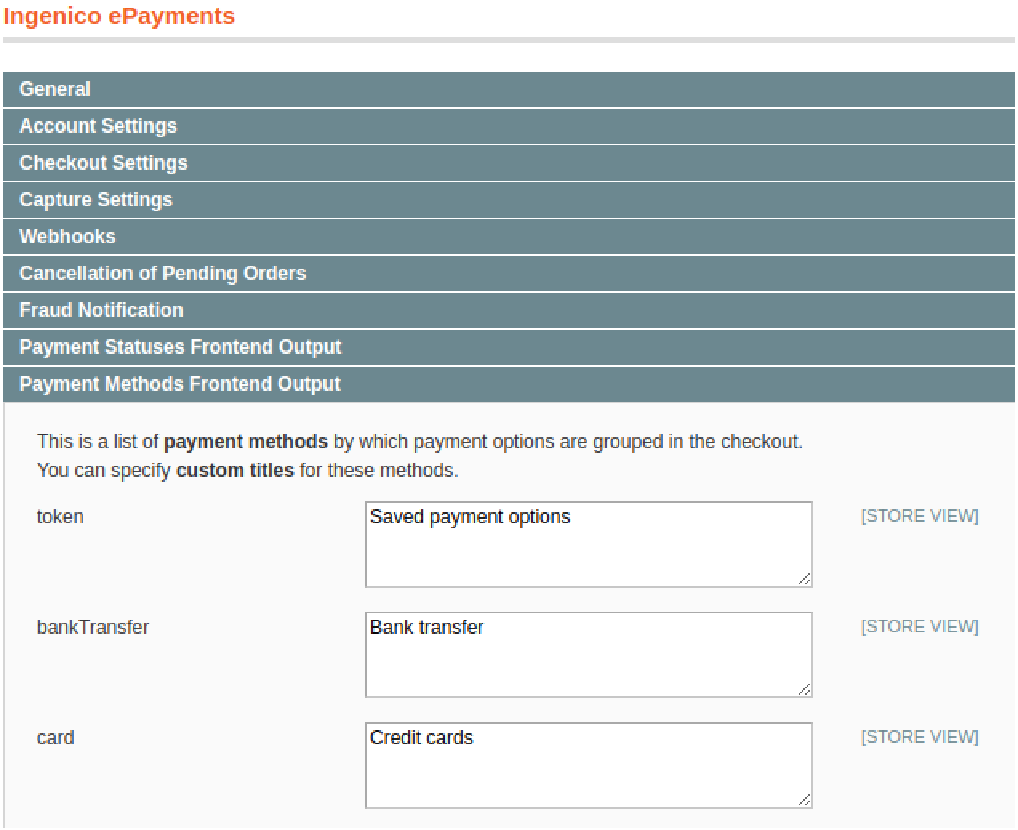
https://developer.globalcollect.com/documentation/api/server/#__merchantId__payments__paymentId__get

3.3.9 Payment methods frontend output

In the section *Payment methods* you will find a list of payment methods groups and their display name in the checkout.

These definition texts will be displayed to the customer for grouping the offered payment methods by type in the Magento® 1 checkout.

You can modify the payment method group label texts on the Store View level according to your needs.



Ingenico ePayments

General
Account Settings
Checkout Settings
Capture Settings
Webhooks
Cancellation of Pending Orders
Fraud Notification
Payment Statuses Frontend Output
Payment Methods Frontend Output

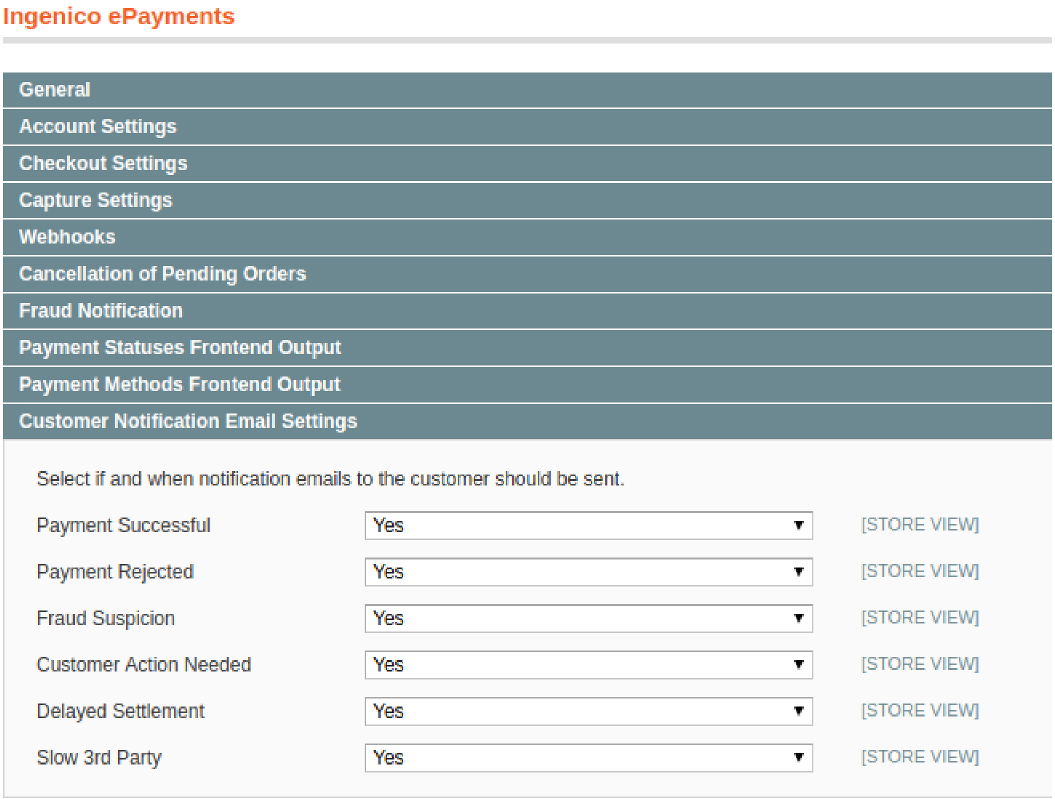
This is a list of **payment methods** by which payment options are grouped in the checkout.
You can specify **custom titles** for these methods.

token	Saved payment options	[STORE VIEW]
bankTransfer	Bank transfer	[STORE VIEW]
card	Credit cards	[STORE VIEW]

3.3.10 Customer notification Email Settings

You can configure when to notify a customer about the status of their payment in the *Customer notification Email Settings* section.

If one of the options is set to *Yes*, the customer will automatically receive a notification email when their payment gets the corresponding status.



Ingenico ePayments

General
Account Settings
Checkout Settings
Capture Settings
Webhooks
Cancellation of Pending Orders
Fraud Notification
Payment Statuses Frontend Output
Payment Methods Frontend Output
Customer Notification Email Settings

Select if and when notification emails to the customer should be sent.

Payment Successful	Yes ▼	[STORE VIEW]
Payment Rejected	Yes ▼	[STORE VIEW]
Fraud Suspicion	Yes ▼	[STORE VIEW]
Customer Action Needed	Yes ▼	[STORE VIEW]
Delayed Settlement	Yes ▼	[STORE VIEW]
Slow 3rd Party	Yes ▼	[STORE VIEW]

4 Workflow and features

4.1 Order workflow

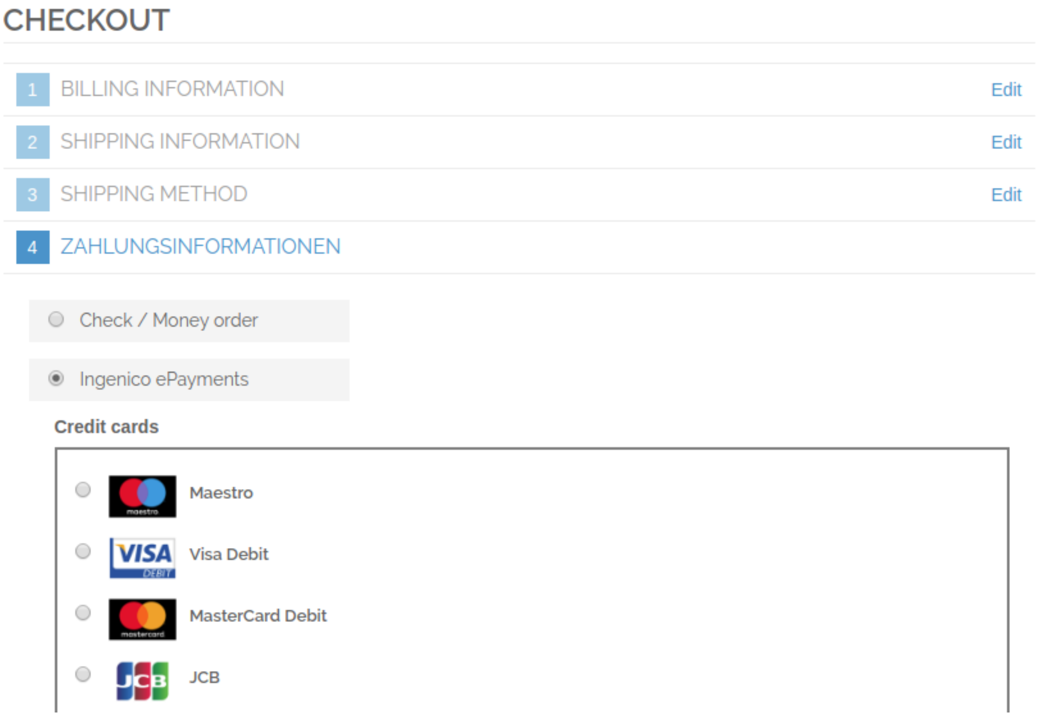
This section describes the standard workflow of an order when using Ingenico ePayments. Special cases and actions will be documented in their respective chapters.

4.1.1 Order placement

If the Ingenico ePayments payment method is enabled in the section Account Settings, the shop customer can select the payment method in the checkout.

The name of the payment method can be configured in the General Settings section.

If this payment method is selected, it offers a dynamic list of payment products, depending on the customer's billing address country, order amount, and the configured Ingenico ePayments endpoint.



CHECKOUT

1 BILLING INFORMATION Edit

2 SHIPPING INFORMATION Edit

3 SHIPPING METHOD Edit

4 ZAHLUNGSINFORMATIONEN

○ Check / Money order

● Ingenico ePayments

Credit cards

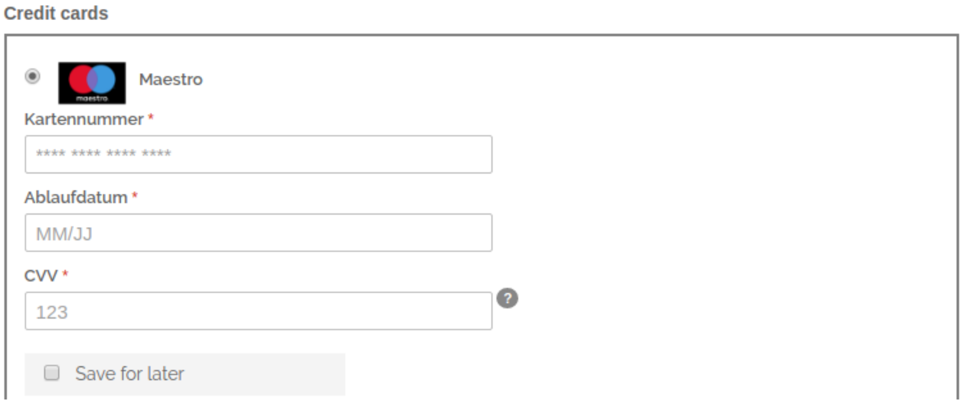
○ Maestro

○ Visa Debit

○ MasterCard Debit

○ JCB

If inline payments has been enabled in the Checkout Settings, depending on the payment product, input fields will show up upon selection.



Credit cards

● Maestro

Kartennummer *

**** **** **** ****

Ablaufdatum *

MM/JJ

CVV *

123

□ Save for later

The entered data gets sent to your Magento® 1 instance as encrypted string and will be forwarded to the Connect Server API upon order placement.

After placing an order with the Ingenico ePayments payment method, the customer is either redirected to the checkout success page or to an external page to enter the payment information. The external page is hosted by Ingenico ePayments.

Afterwards, the customer is redirected back to Magento® 1 and will see the status message from the Payment statuses frontend output section, depending on the status of the payment.

If everything went right, the payment status will be `PENDING_CAPTURE` or `PENDING_APPROVAL` in case of `Delayed settlement` on Capture Settings. Otherwise `CAPTURE_REQUESTED` or `CAPTURED` indicate a successful transaction.

The next section explains how to deal with those statuses.

Styling the external payment page

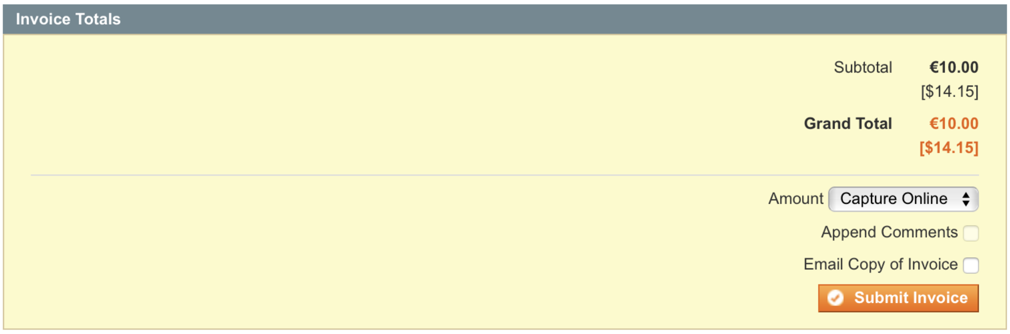
If you want to customize the look and feel of the external payment page, have a look at Ingenico's Hosted Payment Page documentation.

4.1.2 Capture (create invoice)

If you are using the default *Delayed Settlement* capture mode (see section Capture Settings), orders that are processed with Ingenico ePayments have to be manually captured (approved) by the merchant.

This is done by creating an invoice in Magento® 1.

> **Important: use *online* capture**
>
> Make sure that *Capture Online* is selected in the *Amount* box when creating the invoice. Otherwise, the capture will not be registered by the Ingenico ePayments platform.
>
> **If you don't use online capture, the payment flow will be interrupted!**



Invoice Totals

Subtotal €10.00 [$14.15]

Grand Total **€10.00** **[$14.15]**

Amount: Capture Online

Append Comments □

Email Copy of Invoice □

Creating the invoice in Magento® 1 (with *capture online*) starts the capturing process on the Ingenico ePayments platform. However, the funds will most likely not be transferred immediately, which means:

- the order state will be `PAYMENT_REVIEW`,
- the invoice status will be `PENDING`.

This will change after some time, as soon as Ingenico ePayments reports the capture as complete. When the payment status is set to `CAPTURED` by Ingenico, these things will happen:

- the order state will be set to `PROCESSING`,
- the invoice will be marked as `PAID`,
- the funds are transferred to the merchant's account.

From now on, it is possible to perform a *Refund*. This is explained in the next section.

Refresh payment status

You can quickly "pull" the current payment status from the Ingenico ePayments platform using the *Refresh Payment Status* button on the order details page. See also the section Order update.

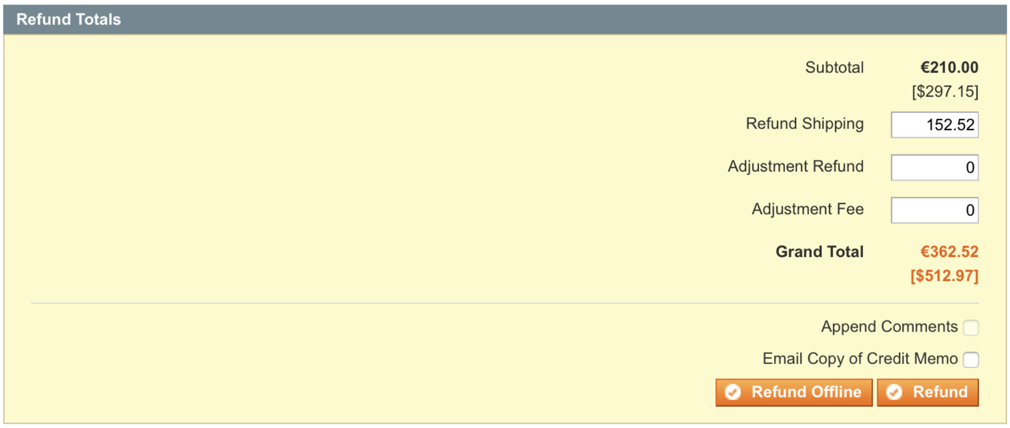
4.1.3 Refund (create creditmemo)

You can trigger refunds on the Ingenico ePayments platform using the Magento® 1 standard refund function.

To see if a payment can be refunded, you can check its Transaction information. Look at the value *isRefundable* in the table.

Follow these steps to make a refund:

- Go to the order details page.
- **Important:** You must open the invoice you want to refund.
- Then select *Creditmemo* from the top menu.
- Adjust the refund amount if needed, and select *Refund* at the bottom of the page.



Refund Totals

Subtotal	**€210.00** [$297.15]
Refund Shipping	152.52
Adjustment Refund	0
Adjustment Fee	0
Grand Total	**€362.52** **[$512.97]**

Append Comments □

Email Copy of Credit Memo □

Important: use *online* refunds

Do **not** choose *Refund Offline* when creating a credit memo. Only online refunds will be registered with the ePayments platform.

If you select *Refund offline*, the buyer will not get their money back.

Similar to captures, refunds will not be processed immediately. There will be a slight time delay.

Refunds through the GlobalCollect endpoint must be explicitly approved (confirmed) on the Credit Memo details page.

As long as the refund is in status `PENDING`, it can be canceled on the Credit Memo details page.

4.1.4 Cancel orders

Orders that were created using the Ingenico ePayments module can be cancelled with the Magento® 1 standard cancel function.

If possible, this will cancel the order on the Ingenico ePayments platform as well.

To see if a payment or refund can be cancelled, you can check its Transaction information. Look at the value *isCancellable* in the table.

Denying a *Suspected Fraud* order will also result in a cancellation, as explained in the section Fraud handling.

Difference between Cancel and Void

For details on what *Cancel* and *Void* mean, have a look at the section Difference between Cancel and Void operations.

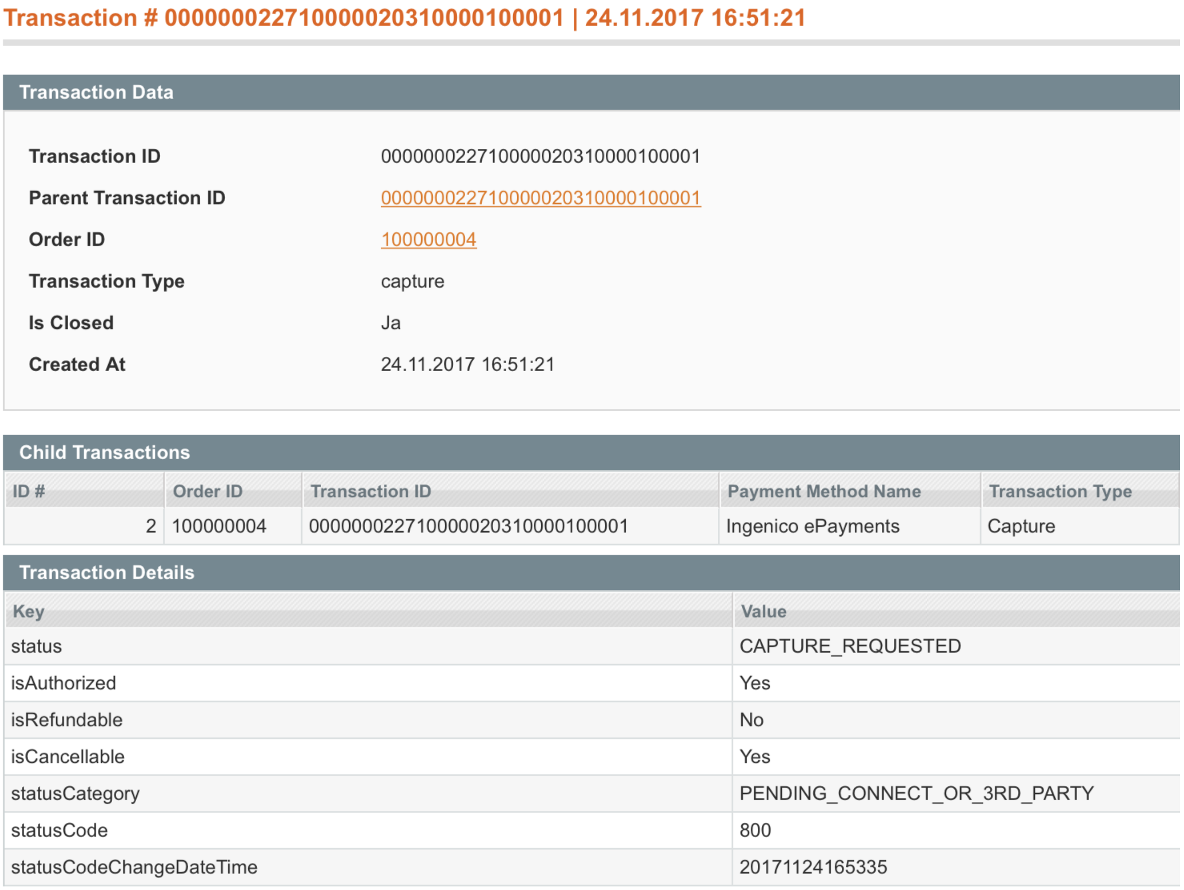
4.2 Additional features and tips

4.2.1 Transaction information

You can look up detailed information about the status of a payment in the Transaction Details:

- Open an order.
- Select *Transactions* from the sidebar.
- Click on a transaction entry in the list.

This shows in-depth details about the transaction status. The data received from the Ingenico ePayments platform is shown in the *Transaction Details* table.



Transaction # 000000022710000020310000100001 | 24.11.2017 16:51:21

Transaction Data

Transaction ID	000000022710000020310000100001
Parent Transaction ID	000000022710000020310000100001
Order ID	100000004
Transaction Type	capture
Is Closed	Ja
Created At	24.11.2017 16:51:21

Child Transactions

ID #	Order ID	Transaction ID	Payment Method Name	Transaction Type
2	100000004	000000022710000020310000100001	Ingenico ePayments	Capture

Transaction Details

Key	Value
status	CAPTURE_REQUESTED
isAuthorized	Yes
isRefundable	No
isCancellable	Yes
statusCategory	PENDING_CONNECT_OR_3RD_PARTY
statusCode	800
statusCodeChangeDateTime	20171124165335

Depending on whether you created an order using the GlobalCollect or Ogone platform, you might see multiple transactions for each stage of the payment, or just one *Authorization Transaction* that holds the current status of the payment.

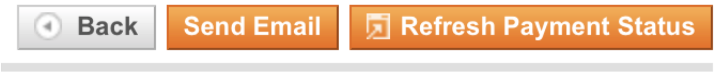
Refunds create separate transactions which show the current status of the refund.

4.2.2 Order update

A payment or refund status can get updated automatically via Webhooks. Alternatively, there are these additional ways to get status updates from the Ingenico ePayments platform:

4.2.2.1 Update an order manually

Click the *Refresh Payment Status* button at the top of the order details view. You will see a message if any changes were found, and the order status will be updated accordingly.



4.2.2.2 Update order automatically

A *Cronjob* is executed every 5 minutes and will query the Ingenico ePayments platform for updates to your orders. The process behind this is similar to the manual order update.

You can configure the Magento® 1 Cron functionality here:

```
System → Advanced → System → Cron (Scheduled Tasks)
```

This also lets you configure a log file that will be stored in the Magento® 1 folder `var/log/`. The log shows information about past automatic order updates.

The Cronjob can also cancel unpaid orders that have not been updated in a certain time. See section Cancellation of Pending Orders on how to configure this feature.

4.2.3 Fraud handling

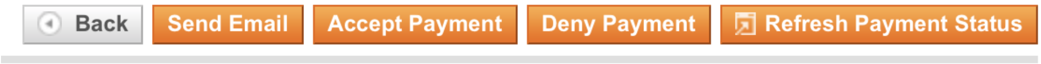
Ingenico ePayments offers automatic Fraud Detection.

If Ingenico has detected a possible fraud:

- The transaction will show a `FRAUD_SUSPECTED` order status.
- A Fraud notification email will be sent to the fraud manager.

Payments that are in *Suspected Fraud* can only be processed further after being *approved* or *denied* by the merchant.

To approve or deny a payment, go to the order details page and use the buttons in the top menu.



The result will be:

- A denied payment will cancel the order, as explained in the section Cancel orders.
- An approved payment will make the order ready for Capture (create invoice).

4.2.4 E-Mail Notifications

You can configure if and when to inform a customer about the order status.

This can be used to keep customers up-to-date and manage their expectations. Please see the section Customer notification Email Settings on how to use this feature.

5 Uninstalling or disabling the module

To **uninstall** the module, follow these steps:

1. Delete all module files from your file system.
2. Remove the module entry `netresearch_epayments_setup` from the table `core_resource`.
3. Remove all module entries in path `ingenico_epayments/` from the table `core_config_data`.
4. Flush the caches afterwards.

If you only want to **disable** the module without uninstalling it, edit the file

```
app/etc/modules/Netresearch_Epayments.xml
```

and set this XML-node from *true* to *false*:

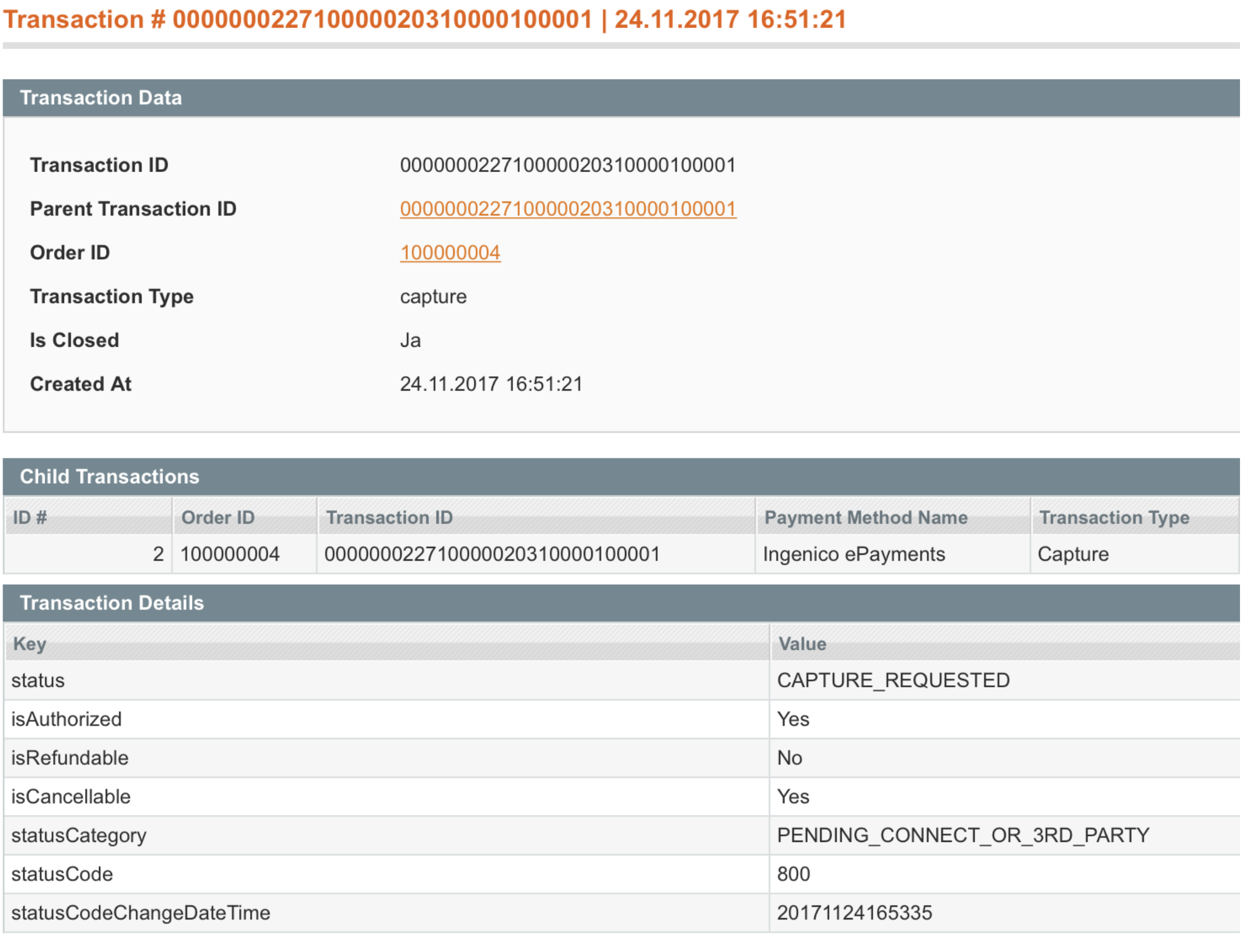
```
config/modules/Netresearch_Epayments/active
```

6 Troubleshooting

6.1 Payment status not accurate

If you believe that the payment status of an order isn't correct or up-to-date, you should first try to refresh the order status with the *Refresh Payment Status* button (see section Order update).

If you need more information for debugging the problem, you can check the Transaction information from the order details page. Select a transaction and look at the *Transaction Details* received from the Ingenico ePayments platform.



Transaction # 000000022710000020310000100001 | 24.11.2017 16:51:21

Transaction Data

Transaction ID	000000022710000020310000100001
Parent Transaction ID	000000022710000020310000100001
Order ID	100000004
Transaction Type	capture
Is Closed	Ja
Created At	24.11.2017 16:51:21

Child Transactions

ID #	Order ID	Transaction ID	Payment Method Name	Transaction Type
2	100000004	000000022710000020310000100001	Ingenico ePayments	Capture

Transaction Details

Key	Value
status	CAPTURE_REQUESTED
isAuthorized	Yes
isRefundable	No
isCancellable	Yes
statusCategory	PENDING_CONNECT_OR_3RD_PARTY
statusCode	800
statusCodeChangeDateTime	20171124165335

6.2 Check log file

If you enabled logging in section Account Settings, you can download the log file using the link:

```
System → Configuration → Ingenico ePayments → Account Settings → Log All Requests → Download log file
```

The log file contains all messages to and from the Ingenico servers and may help finding the cause.

6.3 Technical support

In case of questions or problems, have a look at the Support Portal (FAQ):

http://ingenico.support.netresearch.de/

If the problem cannot be resolved, you can contact the Ingenico support team:

https://www.ingenico.com/epayments/support