

Ingenico ePayments (Ogone) extension for Magento® 2: Setup guide

This document covers the **installation, setup, and configuration** of the Ingenico ePayments (Ogone) extension.

Installation, setup, and configuration

1 Disclaimer	4
2 Requirements	5
2.1 Magento®	5
2.2 PHP	5
2.3 Ingenico ePayments (Ogone) contract	5
3 Hints and general information	6
3.1 Language support	6
4 Installation	7
4.1 Before installation	7
4.2 Installing the module	7
4.2.1 Installing from package using Composer	8
4.2.2 Installing from repository using Composer	9
5 Configuration	10
5.1 Ingenico ePayments (Ogone) backoffice	10
5.1.1 Payment methods	10
5.1.2 User management	10
5.1.3 Technical information	11
5.2 Magento® admin panel	13
5.2.1 Payment Services	13
5.2.2 Payment Methods (general)	15
5.2.3 Open Invoice NL	15
5.2.4 Open Invoice AT via Klarna	15
5.2.5 Credit / debit card	16
5.2.6 Alias payment method	16
5.2.7 Direct Debit	16
5.2.8 PayPal with Seller Protection	17
5.2.9 InterSolve	17
5.2.10 PayPerMail	17
5.2.11 Flex payment method	17
5.3 Device fingerprinting	18
5.3.1 Integration via JavaScript Hook	18
5.3.2 Integration via custom call to provided Controller	18
5.4 Multistore setup	19
5.4.1 Example: Multiple domains, one PSPID	20
5.4.2 Example: Multiple domains, multiple PSPIDs	20

5.4.3 Example: Store Code in URL, one PSPID	21
5.4.4 Example: Store Code in URL, multiple PSPIDs	21
6 Technical support	22

1 Disclaimer

All trademarks, service marks and trade names referenced in this material are the property of their respective owners. The information contained herein is provided as a courtesy and is for general informational purposes only. This information is not intended to be a complete description of all applicable rules, policies and procedures. The matters referenced are subject to change from time to time, and individual circumstances may vary.

Ingenico ePayments (Ogone) shall not be responsible for any inaccurate or incomplete information. The information contained herein includes, among other things, a compilation of information received from third parties, and any such information is and shall remain the property of its respective owner. Nothing contained in this presentation is intended to supplement, amend or modify any applicable contract, rule or regulation.

Ingenico ePayments (Ogone) has given extensive attention to the content of this information but makes no warranties or representations about the accuracy or completeness of it. Neither Ingenico ePayments (Ogone) nor any of its affiliates shall be liable for any costs, losses and/or damages arising out of access to or use of any content of this document. Because of the complexity of the process and the right of Banks to alter conditions, this document can only serve as a description and is subject to further modifications.

The extension referred to in this document was developed as a generic solution. Ingenico ePayments (Ogone) shall not be responsible for any damages that are caused by the extension. In the event that the extension is modified by a user in any way Ingenico ePayments (Ogone) shall not be responsible for any damages that are caused by the modified extension. The description of the extension in this document is provided for convenience purposes only, and Ingenico ePayments (Ogone) makes no warranties or representations about the use or operation of the extension. Neither Ingenico ePayments (Ogone) nor any of its affiliates shall be liable for any costs, losses and/or damages arising out of access to or use of the extension.

2 Requirements

The following requirements must be met for the smooth operation of the extension.

2.1 Magento®

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

- Community Edition 2.1, 2.2, 2.3
- Enterprise Edition 2.1, 2.2, 2.3

2.2 PHP

These PHP versions are supported by the extension:

- PHP 5.6, 7.0, 7.1, 7.2

PHP versions

Please also note the PHP requirements of Magento® itself:

- <https://devdocs.magento.com/guides/v2.2/install-gde/system-requirements-tech.html>

2.3 Ingenico ePayments (Ogone) contract

To use this extension, you need an active Ingenico ePayments (Ogone) account (PSPID).

If in doubt, please contact the Ingenico ePayments (Ogone) Customer Care team for more information.

3 Hints and general information

3.1 Language support

The module supports the locale `en_US`. Translations are stored in CSV translation files and can therefore be modified by third-party modules.

4 Installation

4.1 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.

4.2 Installing the module

Please note

Questions about setting up and using Composer in Magento® are not covered by our technical support. Please refer to these official documentations:

<https://getcomposer.org/doc/01-basic-usage.md>

<https://devdocs.magento.com/guides/v2.2/comp-mgr/install-extensions.html>

4.2.1 Installing from package using Composer

Follow these steps:

- Copy the TGZ or ZIP package into a folder that is accessible by Composer. **Do not unzip the package!**
- On the command line, navigate into the root folder of your Magento® installation.
- Execute the following commands:

```
composer config repositories.ingenico artifact /path/to/folder/with/package/  
composer require ingenico/ingenico_epayments_ogn2  
composer update
```

Note: The path to the package in the above example is absolute, i.e. starting at the server (Linux) root. The path must point to the folder, **not** to the package itself. Omit the filename!

- Check if any errors occurred at this point, and resolve them. See the Composer documentation: <https://getcomposer.org/doc/articles/troubleshooting.md>
- Then execute the following commands:

```
php bin/magento module:enable Ingenico_OPS Netresearch_OPS --clear-static-content  
php bin/magento setup:upgrade  
php bin/magento cache:flush  
php bin/magento setup:di:compile  
php bin/magento setup:static-content:deploy en_US en_GB fr_FR de_DE it_IT es_ES
```

Note: After the command "setup:static-content:deploy", put the list of your shop's locales. The above is just an example (but it should cover most scenarios).

- Check if any errors occurred.
- Then proceed with the [Configuration](#).

4.2.2 Installing from repository using Composer

If you have received the module from the Magento® Marketplace, you can access their repository to install the module with Composer.

Follow these steps:

- Get your authentication keys. These are needed to access the Magento® repository, see also <https://devdocs.magento.com/guides/v2.2/install-gde/prereq/connect-auth.html>
- On the command line, navigate into the root folder of your Magento® installation.
- Execute the following commands:

```
composer require ingenico/ingenico_epayments_ogn2
composer update
```

- These steps are also explained in the official Magento® guide for Composer installations: <https://devdocs.magento.com/guides/v2.2/comp-mgr/install-extensions.html>
- Check if any errors occurred at this point, and resolve them. See the Composer documentation: <https://getcomposer.org/doc/articles/troubleshooting.md>
- Then execute the following commands:

```
php bin/magento module:enable Ingenico_OPS Netresearch_OPS --clear-static-content
php bin/magento setup:upgrade
php bin/magento cache:flush
php bin/magento setup:di:compile
php bin/magento setup:static-content:deploy en_US en_GB fr_FR de_DE it_IT
```

- **Note:** After "setup:static-content:deploy", put the list of your shop's locales. The above is just an example (but it should cover most scenarios).
- Check if any errors occurred.
- Then proceed with the [Configuration](#).

5 Configuration

5.1 Ingenico ePayments (Ogone) backoffice

Login to your Ingenico ePayments (Ogone) backoffice to configure the following settings.

5.1.1 Payment methods

Depending on the Ingenico ePayments (Ogone) subscription you have selected, it might be necessary to activate additional mandatory options in your Ingenico ePayments (Ogone) account:

- OPC
- DirectLink (New Payment)
- DirectLink (Maintenance)
- D3D
- Alias Manager (see also the User Guide)

Please check if these options are activated in "Configuration -> Account -> Your options".

If this is not the case, please contact the Ingenico ePayments (Ogone) Customer Care team for activation.

5.1.2 User management

The Magento® extension needs an API user to communicate with the Ingenico ePayments (Ogone) platform.

To create the API user in the Ingenico ePayments (Ogone) backoffice:

- Go to "Configuration -> Users".
- Select "New User".
- Create a USERID (for example "APIUSER").
- Fill in the name and an existing email address.
- The field "External USERID" can be left empty.
- Select the profile "Administrator".
- Select the access type "API only".
- Click the "Save" button.
- Document the USERID and the password which you have created.

API user and password

The USERID and password may not contain any special characters or umlauts. Use only letters and numbers.

5.1.3 Technical information

Configure the section "Configuration -> Technical Information" in the Ingenico ePayments (Ogone) backoffice as follows. Any settings that are not mentioned here should be left as they are.

On the tab **Global Transaction parameters**:

- In the "Payment retry" section, change the value if you want to, e.g. to "2".
- In the "Processing for individual transactions" section, select "Always online (Immediate)".
- The setting "Default operation code" is not important. This is overridden by the setting "Payment Action" in Magento®, section [Payment Services](#).
- Click the "Save" button.

On the tab **Global security parameters**:

- Set the "Hashing algorithm" to "SHA-256" or "SHA-512". This must be identical to the setting in Magento® in [Payment Services](#). Using "SHA-1" is not recommended for security reasons (but technically still works).
- Set the "Character encoding" to "UTF-8".
- In the (optional) section "Template", set "Enable JavaScript check on template" to "No".
- Click the "Save" button.

On the tab **Data and origin verification**:

- In the sections "Checks for e-Commerce & Alias Gateway" and "Checks for DirectLink and Batch (Automatic)", create a "SHA-IN pass phrase".
- Click the "Save" button.

SHA-IN configuration

The SHA-IN passphrase must be identical in all fields, can only be up to 20 characters long, and must contain only numbers and letters (no special characters or umlauts).

It also should be identical to the SHA-OUT passphrase (see below).

On the tab **Transaction feedback**:

- Enable the option "I would like to receive transaction feedback parameters on the redirection URLs".
- Disable the option "I would like Ingenico ePayments (Ogone) to display a short text to the customer...".
- In the section "Direct HTTP server-to-server request", select "Online but switch to a deferred request when the online requests fail".
- In the section "URL of the merchant's post-payment page", enter the value "https://YOUR_SHOP_SECURE_BASE_URL/ops/api/postBack" in both fields.
- Set "Request method" to "POST".
- In the section "Dynamic e-Commerce parameters", select (enable) all available parameters **except** "CN", "ECOM_BILLTO..." and "ECOM_SHIPTO...", because these can cause transmission errors.
- In the section "Alias gateway and Tokenization - Dynamic parameters", select (enable) all parameters.
- In the section "All transaction submission modes", create a "SHA-OUT pass phrase" (only up to 20 characters, only numbers and letters, no special characters or umlauts, and identical to the SHA-IN passphrase).
- In the section "HTTP request for status changes", select "For each offline status change ...".
- Enter the value "https://YOUR_SHOP_SECURE_BASE_URL/ops/api/directLinkPostBack" into the field.
- In the section "DirectLink - Dynamic parameters", select (enable) all parameters **except** "CN", "ECOM_BILLTO..." and "ECOM_SHIPTO...".
- Click the "Save" button.

SHA-OUT configuration

The SHA-OUT passphrase can only be up to 20 characters long, and contain only numbers and letters (no special characters or umlauts).

It also should be identical to the SHA-IN passphrase (see above).

On the tab **Transaction e-mails** (optional):

- In the section "E-mails to the merchant", fill in you email address.
- Select "Yes, for all transaction submission modes."
- Select "Yes, for each offline status change..."

On the tab **Test info**:

- Select "I would like to simulate transaction results based on the card number".

5.2 Magento® admin panel

Log in to your Magento® admin panel to configure the following settings.

5.2.1 Payment Services

Configure the section "Configuration -> Sales -> Payment Services -> Ingenico ePayments (Ogone) Account" in the Magento® admin panel as follows:

- In the field "PSPID", enter the PSPID of your Ingenico ePayments (Ogone) account.
- Set "Hash algorithm" to the same value as in the Ingenico ePayments (Ogone) backoffice.
- Set "SHA-IN" / "SHA-OUT Passphrase" to the same value as in the Ingenico ePayments (Ogone) backoffice. **Both should be identical!** See also [Technical information](#).
- Set "Environment" to either "Test" or "Prod". If you select "Custom", you must enter the correct gateway URLs in the fields below.
- In the fields "API User" / "Password", enter the credentials which you created in the Ingenico ePayments (Ogone) backoffice. See also [User management](#).
- The option "Payment Action" will override the setting in the Ingenico ePayments (Ogone) backoffice. Set this to:
 - "Direct Sale" for automated processing of payments and invoice creation, or
 - "Authorize" if you want to trigger the capture manually by creating the invoice.
- Set "Send Invoice to Customer" to "Yes" if you want to automatically trigger the invoice emails.
- The option "Payment Template" controls the styling of the payment page for redirect payment methods:
 - "Redirect mode": The customer will get redirected after the checkout to Ingenico ePayments (Ogone) to enter the payment details. **Note:** If this is selected, additional fields will be shown below to style the payment page (fonts, colors, additional buttons, payment method layout, etc.).
 - "Internal shop template": The customer will be redirected to the Ingenico ePayments (Ogone) page with the look and feel of your shop.
 - "iFrame mode": **Not recommended, see below.** Payment methods are shown in an iFrame in the checkout. iFrame can be styled through the parameters below. This does not affect Flex Checkout (HTP) credit / debit card payments. Those are configured directly in the respective payment method.
 - "Dynamic template": The customer will be redirected to the Ingenico ePayments (Ogone) page. The look and feel of that page will be defined by a dynamically loaded template file. The origin can be defined in the field "Template Identifier/Url" below.

Payment Template: iFrame mode

Using this mode is **not recommended** due to incompatibilities with certain providers (e.g. PayPal, Sofort.com). Their payment pages do not work properly inside iFrames.

[Credit / debit card](#) payment methods can still be shown in an iFrame, regardless of the above setting. The configuration for that is made in the respective payment method in Magento®.

- Set "Automatic Payment Reminder" to "Yes" if you want to send a reminder email to the customer if the order is unpaid (status 0, 1, 2) after a defined time. The number of days can be set in the field "Reminder Days" below.
- Set "Automatic Payment Cancellation" to "Yes" if you want to cancel unpaid (stale) orders automatically (status 0, 1, 2) after a defined time. The number of days can be set in the field "Cancellation Days" below. The setting "Automatic Payment Reminder" will be taken into account to prevent too early cancellations.
- Set "Show Quote ID in the order grid" to "Yes" if you want to see the Magento® Quote ID in the order list.
- Set "Submit extra parameters" to "Yes" if you want to transmit additional parameters to Ingenico ePayments (Ogone). **Note:** Enable this only when using Fraud Detection or [PayPal with Seller Protection](#)!
- Set "Resend payment information identity" and "Resend payment information template" as desired for payment reminder emails to the customer.
- Set "PayPerMail Email Template" as desired for [PayPerMail](#) emails.
- Set "Enable device fingerprinting" if you want to transmit hashed customer data to Ingenico ePayments (Ogone). **Note:** This requires the explicit consent from the customer! See also [Device fingerprinting](#).
- Set "Timeout for transactions" to the desired number of seconds.
- Set "Credit/Debit card Split" to "Yes" if you want separate payment methods being displayed for credit and debit cards. **Note:** This requires additional configuration in your Ingenico ePayments (Ogone) account, otherwise the **transactions will fail!**
- Set "Debug" to "Yes" to write the log file. Note that the Magento® logging may also have to be activated.

5.2.2 Payment Methods (general)

Configure the section "Configuration -> Sales -> Payment Methods" in the Magento® admin panel as follows:

- Activate the payment methods which you want to offer, and which are also enabled for you at Ingenico ePayments (Ogone). To activate additional payment methods, please contact the Ingenico ePayments (Ogone) Customer Care team.
- You can configure the applicable countries for each payment method in Magento®. This controls if the payment method will be shown in the checkout, based on the customer's billing country.
- Each payment method allows storing a logo for the checkout. The position of the logo is also configurable. If you don't want to show a logo, set "Logo position" to "Hidden". A set of default logos is included.

Some payment methods require a more complex configuration. This is explained below.

5.2.3 Open Invoice NL

Configure the section "Configuration -> Customers -> Customer Configuration -> Name and address options" as follows:

- Set "Show Tax/VAT number" to "Required" (only for Open Invoice NL).
- Additional information (gender, date of birth) will be automatically requested in the checkout, if needed.

Note: If AfterPay is the acquirer, it is **not** possible to perform partial captures with Open Invoice NL.

5.2.4 Open Invoice AT via Klarna

Configure the section "Configuration -> Sales -> Payment methods -> Ingenico ePayments (Ogone) Open Invoice AT" as follows:

- Set "Title" to "Kauf auf Rechnung - zahlen in 14 Tagen".
- Set "Request data encoding" to "Other". UTF-8 is **not** supported in this particular case.
- Set "Invoice terms title" to "Rechnungsbedingungen".
- Set "Invoice terms URL" to:
https://cdn.klarna.com/1.0/shared/content/legal/terms/123/de_at/invoice?fee=0#
- Upload the logo to:
<https://developers.klarna.com/en/at+php/kpm/logos>
- Magento® should not send invoices on capture / refund to the consumers. Klarna will send the invoice. During the capture / refund, the setting "Email Copy of Invoice" should be unchecked.
- A partial capture or partial refund via Magento® is not possible with Klarna. These actions can only be executed via the Ingenico ePayments (Ogone) backoffice.
- Additional information (gender, date of birth) will be automatically requested in the checkout, if needed.

5.2.5 Credit / debit card

The configuraton is made in the section "Configuration -> Sales -> Payment Methods". There are several payment methods available:

- Ingenico ePayments (Ogone) Cc (Directlink)
- Ingenico ePayments (Ogone) Dc (Directlink)
- Other Ingenico ePayments (Ogone) CC brands (Redirect)
- Other Ingenico ePayments (Ogone) DC brands (Redirect)

Enable the payment method(s), depending on the desired behavior / customer experience:

- "Directlink": The payment method will be displayed in an iFrame inside the checkout, without redirecting the customer to an external payment page. **This is recommended.**
- "Redirect": The customer will get to an external payment page after finishing the checkout.

Each payment method has the following options, which can be set individually:

- In the box "Card types", select (enable) the brands you want to accept.
- Set "Enable 3D secure" to "Yes" to redirect the customer to an external page for entering the 3DS code (if supported by the card). To enter the 3DS code, the customer will be redirected to an external page, even when using the iFrame (Directlink) in the checkout.
- Set "Enable for backend" to "Yes" if you want to use the payment method for admin orders, too (MoTo).
- Set "Enable Alias Manager" to "Yes" if you want to allow the customer to save their payment information for later reuse.

The feature "Alias Manager" needs to be enabled in your Ingenico ePayments (Ogone) account if you want to offer it. If in doubt, please contact the Ingenico ePayments (Ogone) Customer Care team.

5.2.6 Alias payment method

This payment method allows using saved payment information (Aliases). It works in conjunction with the [Credit / debit card](#) and [Direct Debit](#) payment methods.

5.2.7 Direct Debit

You can use this payment method for Austria, Germany, and the Netherlands. The customer can use their account data for payments.

Configure the section "Configuration -> Sales -> Payment methods -> Ingenico ePayments (Ogone) Direct Debit" as follows:

- In the box "Countries", select (enable) the countries for which you want to offer this payment method.
- Set "Enable for backend" to "Yes" if you want to use the payment method for admin orders, too (MoTo).
- Set "Enable Alias Manager" to "Yes" if you want to allow the customer to save their payment information for later reuse.

5.2.8 PayPal with Seller Protection

To use PayPal Seller Protection, make the following configuration:

- Go to "Configuration -> General -> State options -> State is required for" and set the countries for which the state is mandatory, according to the requirements of PayPal Seller Protection.
- Go to "Configuration -> Sales -> Payment services" and set "Submit extra parameters" to "Yes".

5.2.9 InterSolve

The configuraton is made in the section "Configuration -> Sales -> Payment Methods -> InterSolve".

You can add several brands, which are related to specific Ingenico ePayments (Ogone) payment methods. These brands have to be configured in advance in the Ingenico ePayments (Ogone) backoffice before they can be used in your shop.

You can assign a customised "Title" to each "Brand" which is shown to the customer on the checkout page.

5.2.10 PayPerMail

This payment method is only available for admin orders (MoTo).

When it is used, an email with a link is sent to the customer. The customer can follow the link to execute the payment for the order that was created in the Magento® admin panel.

5.2.11 Flex payment method

This is a generic payment method that lets you choose the exact method on Ingenico ePayments (Ogone) side. It can be used to offer payment methods that are not already implemented in the Magento® extension.

The configuration is made in "Configuration -> Sales -> Payment methods -> Ingenico ePayments (Ogone) Flex Payment Method".

- Set "Title" to the value you want to show in the checkout.
- Set "PM" and "BRAND" to the values you received from Ingenico ePayments (Ogone) Customer Care. These values are transmitted in the respective parameters to the Ingenico ePayments (Ogone) platform.

5.3 Device fingerprinting

This feature allows the extension to fingerprint customer devices for Ingenico ePayments (Ogone) to collect and hash data (such as screen resolution, user agent, etc.) into a key. There are legal caveats in some European countries, therefore customers should consent to the collection of these data during the checkout process.

The device fingerprinting can be enabled or disabled per transaction. This is indicated by a special parameter for e-Commerce transactions. For DirectLink transactions, information will have to be collected by tracking pixels on the site.

5.3.1 Integration via JavaScript Hook

The extension provides a JavaScript function to handle customer's consent in the checkout.

Merchants can also add the file *js/netresearch/ops/deviceFingerprinting.js* to other parts of the shop to have access to the functions described below.

This can be done via the following code:

```
consentHandler.toggleConsent(consent, callback);
```

- *Consent*: Boolean and the target state of consent that should be saved in the customer session.
- *Callback*: An optional function that should get called with the saved consent after it has been saved.

The consent handler also has a function of retrieving the current consent status via:

```
consentHandler.getConsent (callback)
```

As with the other function the callback will be called with the result of the request (the current state of consent).

5.3.2 Integration via custom call to provided Controller

Consent can also be indicated by calling a controller directly on the following URL:

http://YOUR_SHOP_URL/ops/device/toggleConsent

The parameter `consent` is expected for the request. It will get casted to boolean and saved to the customer's session. A response of the current consent state as JSON will be for example:

```
{"consent":false}
```

The current state of consent can be viewed here:

http://YOUR_SHOP_URL/ops/device/consent

The pre-built JavaScript functions call these URLs.

5.4 Multistore setup

You can use the extension in Magento® multistore setups with:

- multiple websites
- multiple stores
- multiple domains
- multiple Ingenico ePayments (Ogone) accounts (PSPIDs)

To use multiple Ingenico ePayments (Ogone) accounts, you have to follow the configuration steps mentioned in [Ingenico ePayments \(Ogone\) backoffice](#) and [Magento® admin panel](#) for each Ingenico ePayments (Ogone) account, website, and / or store.

The most important part is to configure the transaction feedback URLs in the Ingenico ePayments (Ogone) backoffice correctly. These are found in "Configuration -> Technical information -> Transaction feedback":

- Direct HTTP server-to-server request (two URL fields)
- HTTP request for status changes (one URL field)

The configuration is clarified in the examples below.

Please note

All websites, stores, and domains must be hosted by the same Magento® instance. The Order ID (Merch Ref) must be unique accross all orders. Otherwise, the error "This order has already been processed" (or similar) can occur.

5.4.1 Example: Multiple domains, one PSPID

The example-multistore consists of two stores which are assigned to different websites (domains). Both use the same account (PSPID):

- Store A operates at <https://your-shop-a.com/>
- Store B operates at <https://your-shop-b.com/>

The configuration for **both** stores in the Ingenico ePayments (Ogone) backoffice is:

- Direct HTTP server-to-server request: <https://your-shop-a.com/ops/api/postBack>
- HTTP request for status changes: <https://your-shop-a.com/ops/api/directLinkPostBack>

The feedback will be processed correctly for store **B** also, even if the response is sent only to the URL of store **A**.

You have to configure the Magento® base URL for Store **B** like this:

- Go to "Configuration -> General -> Web".
- In the section "Unsecure", put "<http://your-shop-b.com/>" in the fields "Base URL" and "Base Link URL".
- In the section "Secure", put "<https://your-shop-b.com/>" in the fields "Base URL" and "Base Link URL".

Leave the default values in the other fields.

5.4.2 Example: Multiple domains, multiple PSPIDs

The example-multistore consists of two stores which are assigned to different websites (domains). It uses two separate accounts (PSPIDs):

- Store A operates at <https://your-shop-a.com/>
- Store B operates at <https://www.your-shop-b.com/>

The configuration for store **A** in the Ingenico ePayments (Ogone) backoffice (first PSPID) is:

- Direct HTTP server-to-server request: <https://your-shop-a.com/ops/api/postBack>
- HTTP request for status changes: <https://your-shop-a.com/ops/api/directLinkPostBack>

And the configuration for store **B** in the Ingenico ePayments (Ogone) backoffice (second PSPID) is:

- Direct HTTP server-to-server request: <https://your-shop-b.com/ops/api/postBack>
- HTTP request for status changes: <https://your-shop-b.com/ops/api/directLinkPostBack>

5.4.3 Example: Store Code in URL, one PSPID

The example-multistore consists of two stores which are assigned to the same website (domain). Store Codes are enabled. A single PSPID is used for both stores.

- Store A operates at <https://your-shop.com/shop-a/>
- Store B operates at <https://your-shop.com/shop-b/>

This example is also valid in case both stores were assigned to different websites.

The configuration in the Ingenico ePayments (Ogone) back office is therefore:

- Direct HTTP server-to-server request: <https://your-shop.com/shop-a/ops/api/postBack>
- HTTP request for status changes: <https://your-shop.com/shop-a/ops/api/directLinkPostBack>

The feedback will be processed correctly for store **B** also, even if the response is sent only to the URL of store **A**.

5.4.4 Example: Store Code in URL, multiple PSPIDs

The example-multistore consists of two stores which are assigned to the same website (domain). Store Codes are enabled. Two separate accounts (PSPIDs) are used.

The example is also valid if both stores were assigned to different websites.

- Store A operates at <https://your-shop-a.com/>
- Store B operates at <https://your-shop-b.com/>

The configuration for Store **A** in the Ingenico ePayments (Ogone) backoffice (first PSPID) is:

- Direct HTTP server-to-server request: <https://your-shop-a.com/ops/api/postBack>
- HTTP request for status changes: <https://your-shop-a.com/ops/api/directLinkPostBack>

And the configuration for Store **B** in the Ingenico ePayments (Ogone) backoffice (second PSPID) is this:

- Direct HTTP server-to-server request: <https://your-shop-b.com/ops/api/postBack>
- HTTP request for status changes: <https://your-shop-b.com/ops/api/directLinkPostBack>

6 Technical support

In case of questions or problems, have a look at the Knowledge Base (FAQ). Try entering the error message (or part of it) into the search function.

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

If the problem cannot be resolved, please contact the Ingenico ePayments (Ogone) Customer Care team:

support@ecom.ingenico.com