

Bridge module - Documentation

Sommaire

1. Are you a Bridge customer?	2
2. What is Pay by Account-to-Account Transfer (by Bridge)?	2
3. Installing the Bridge Module	3
4. Configuring the Bridge Module	4
1. Build your Bridge app	7
2. Copy the API keys to your Magento module	10
3. Configure your WebHook	12
5. Use of the module	15

1. Are you a Bridge customer?

You are already a Bridge customer: Go to [chapter 2](#)

Not a Bridge customer?

- [Open an account](#)
- Download and install the Bridge module and start accepting direct payments on your Magento.

2.What is Pay by Account-to-Account Transfer (by Bridge)?

Integrate Bridge, a secure plug-and-play payment solution, to increase your conversion rates at reduced costs. Enjoy easy reconciliation and low fraud rates.

The Bridge payment solution allows you to:

- Offer high payment ceilings

Increase your conversion rates by allowing top carts to pay large sums instantly.

- Take advantage of a low-cost payment solution

Both cost management and cash flow management are simplified. You can take advantage of a more competitive cost than using a credit card by paying a low percentage only on the transactions made on your site.

- Enjoy instant payments

Receive funds in your bank account fast.

- Benefit from irrevocability of payments

Payments are irrevocable. In other words, customers cannot cancel or modify their payments.

- Reduce fraud rates and offer secure payment

With account-to-account transfer, at the time of payment, your customer is authenticated directly to their online account. No sensitive data will be entered, which makes it possible to offer a secure payment process with a limited risk of fraud.

3. Installing the Bridge Module

Composer installation

Install module : composer require bridgepay/bridge-module-magento2 --no-dev

Run : php bin/magento module:enable Bridgepay_Bridge

Run : php bin/magento setup:upgrade

Run : php bin/magento setup:static-content:deploy

Flush caches with : php bin/magento cache:flush

Update with Composer

To update the extension to the latest available version (depending on your composer.json), run these commands in your terminal :

composer update bridgepay/bridge-module-magento2 --with-dependencies --no-dev

php bin/magento setup:upgrade

php bin/magento setup:di:compile

php bin/magento setup:static-content:deploy

php bin/magento cache:flush

Manual installation

Install SDK using Composer : composer require 202ecommerce/bridge-sdk:1.0.0

Download Bridge's module from Magento's Marketplace :

<https://marketplace.magento.com/extensions.html>

Copy / Paste module files to your Magento root directory

Run : php bin/magento module:enable Bridgepay_Bridge

Run : php bin/magento setup:upgrade

Run : php bin/magento setup:static-content:deploy

Flush caches with : php bin/magento cache:flush

Maintenance mode

You may want to enable the maintenance mode when installing or updating the module, especially when working on a production website. To do so, run the two commands below before and after running the other setup commands :

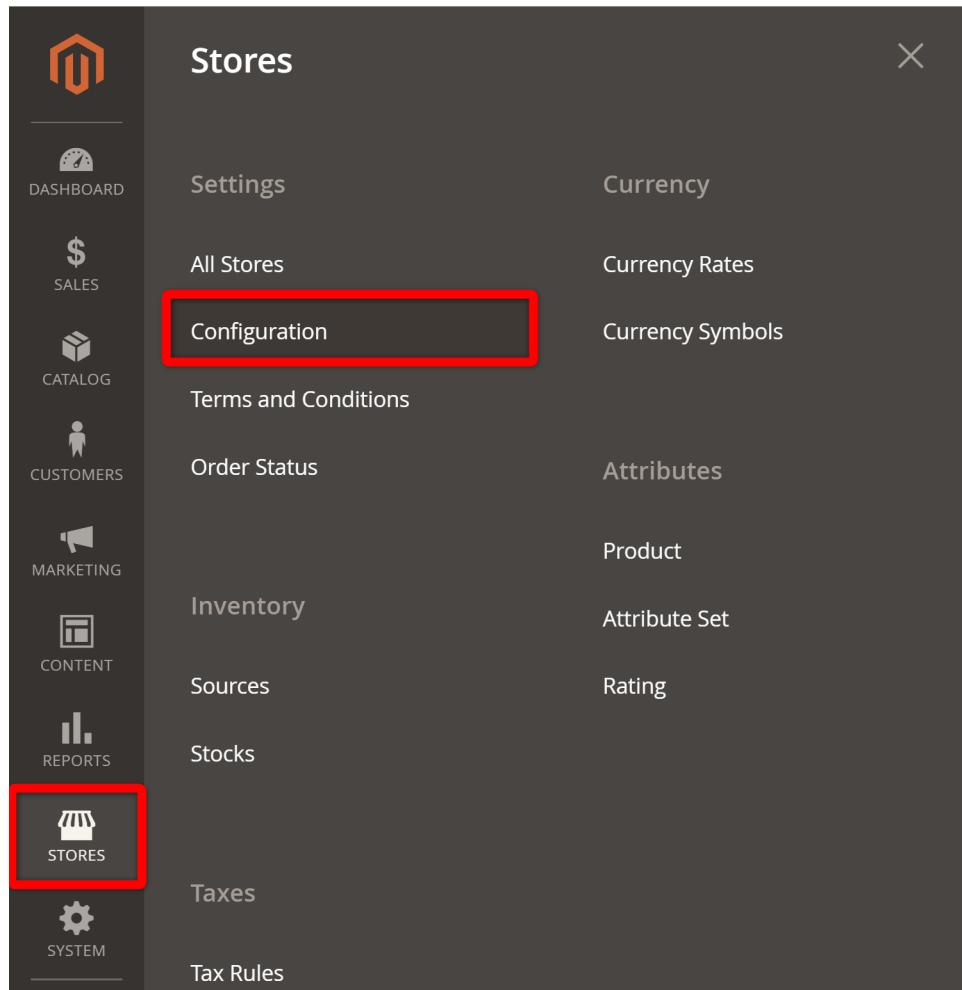
php bin/magento maintenance:enable

Other setup commands

php bin/magento maintenance:disable

4. Configuring the Bridge Module

Once the module is installed by Magento, go to 'Stores > Configuration'



Then select the tab 'Bridge official > Setting up'

The screenshot shows the Magento Configuration interface. On the left is a sidebar with various tabs: DASHBOARD, SALES, CATALOG, CUSTOMERS, MARKETING, CONTENT, REPORTS, STORES, SYSTEM, and FIND PARTNERS & EXTENSIONS. The 'SYSTEM' tab is currently selected. The main area is titled 'Configuration'. At the top, there is a 'Scope:' dropdown set to 'Configuration par défaut' and a help icon. Below this, a list of categories is shown with arrows indicating expandable sections: GENERAL, CATALOG, SECURITY, CUSTOMERS, BRIDGE OFFICIAL, and SALES. The 'BRIDGE OFFICIAL' and 'Setting up' sub-sections are highlighted with red boxes. The 'Setting up' section is expanded, showing its contents.

A status screen allows you to validate the conditions of use of the Bridge module and certain configuration states

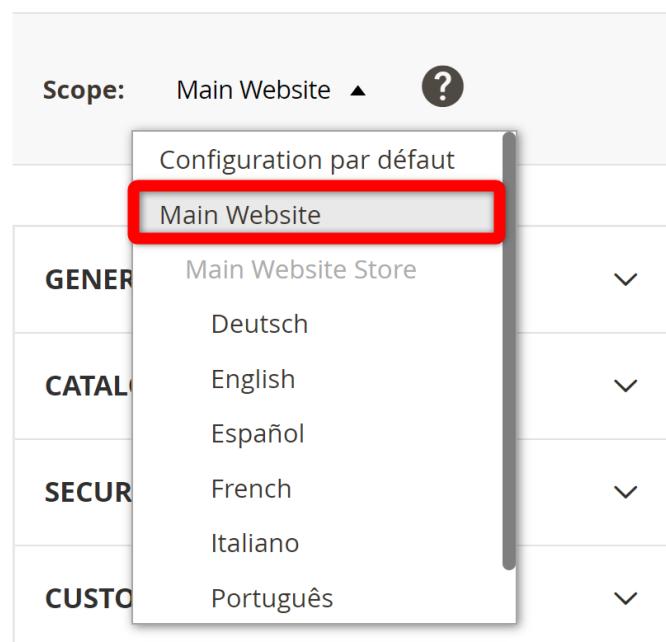
Status

- ✓ CURL - version v7.74.0 OpenSSL/1.1.1n
- ✓ SSL & TLS1.2 - Enabled
- ✗ Connected to bridge API - Error not in website or store context
- ✗ Production environment
- ✗ WebHook contacted

✗ Error not in website or storeview

To configure your module you must change the 'Scope' to 'Website' or 'Storeview'

Configuration



✗ Error in client ID / Secret (not saved)

Once in the right scope you need to get your API Keys.

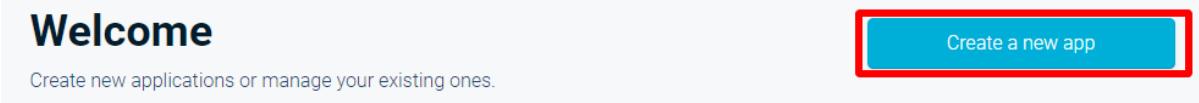
Status

- ✓ CURL - version v7.74.0 OpenSSL/1.1.1n
- ✓ SSL & TLS1.2 - Enabled
- ✗ Connected to bridge API - Error in client ID / Secret (not saved)
- ✗ Production environment
- ✗ WebHook contacted

1. Build your Bridge app

Log in to your Bridge account at <https://dashboard.bridgeapi.io> to configure the elements necessary to configure your module.

Create an app on your Bridge account



Save all the parameters necessary for your application by saving the URL of your website, your bank identifier, etc.

Create your application

1 Bridge products activation

You can create several applications with different products

Payment initiation



Payments

Online payments solution to pay and get paid (compatible with our plugins WooCommerce, PrestaShop...)

[Documentation →](#)



Payments links

Payments links generation to easily get paid without code or through API

[Documentation →](#)

Warning: The Bridge API being protected from intrusions, it is imperative that you specify the IP address(es) of your server(s) that must query the Bridge API remotely. To find the IP address of a server, go to a command prompt and run the following code: ping www.myserver.com

The response to this request returns an IP address to be indicated in the 'Authorized IP addresses' field.

Create your application

Bridge products activation ▼

Payments configuration ▼

3 General settings

These settings apply to all activated products

Internal name

Name of your platform or your online store

This name will never be displayed to your users

Authorized IPs ?

8.8.8.8, 8.8.4.4

Enter your IPs separated by commas

Create your application

2. Copy the API keys to your Magento module

Once your application is configured:

- Retrieve the 'Client ID' and 'Client Secret' API keys in your Dashboard Bridge: App details > Settings
- Copy them into your Magento module
- Click on 'Save'

Note: the API keys are the same in the test environment and in the prod environment, we advise you to first carry out tests then activate the prod mode after having validated the entire payment flow in mode test.

The screenshot shows the Magento Admin Panel under the 'My App' section. On the left sidebar, there are links for 'Preview', 'Settings', 'Payments' (with 'Payments list' and 'Payments analytics'), and 'Webhooks'. The main area is titled 'Settings' and contains a 'Application details' section. It shows 'Client id' (16e61de8b8de4c2aa5a1a8806479052b) and 'Client secret' (redacted). A red box highlights these fields. To the right are 'Copy' and 'Regenerate' buttons. Below this, there's an 'Environment' dropdown set to 'Development' with a note: 'This option defines in which environment your module is configured'. A red arrow points from the 'Client secret' field in the top box down to the 'Client Secret (Sandbox)' field in the bottom box. The bottom box also contains 'Client ID (Sandbox)' (16e61de8b8de4c2aa5a1a8806479052b) and 'Client Secret (Sandbox)' (redacted), with notes: 'This information is located in your Bridge dashboard: 'Settings' > 'General settings''.

After this first step, the module must be connected to the API Bridge

- **'Development'** environment

Status

- ✓ CURL - version v7.74.0 OpenSSL/1.1.1n
- ✓ SSL & TLS1.2 - Enabled
- ✓ Connected to bridge API - 181 banks
- ✗ Production enviroment
- ✗ WebHook contacted

- **'Production'** environment

Status

- ✓ CURL - version v7.74.0 OpenSSL/1.1.1n
- ✓ SSL & TLS1.2 - Enabled
- ✓ Connected to bridge API - 181 banks
- ✓ Production enviroment
- ✗ WebHook contacted

3. Configure your WebHook

The majority of payments will be validated when the customer returns to your Magento store. In some cases, the validation happens in a second time, the order therefore changes to the status of payment accepted thanks to the WebHook.

Get the url of your WebHook in your Magento module

Status

- ✓ CURL - version v7.74.0 OpenSSL/1.1.1n
- ✓ SSL & TLS1.2 - Enabled
- ✓ Connected to bridge API - 181 banks
- ✗ Production environment
- ✗ WebHook contacted

WebHook configuration

<https://bridgemagento.magento.202-dev.co...> 

This URL is needed to connect Bridge to your website. It must be entered in the WebHook section of your Bridge account for the module to work properly.

Go to your Bridge dashboard to create and configure your Webhook. When adding the WebHook, copy the url retrieved from the Magento module in the 'URL' field and give a name to your WebHook. Choose the set of events you want to send to your WebHook. The following events are necessary for the correct functioning of the Magento <> Bridge exchanges:

- payment.transaction.created
- payment.transaction.updated

Module Bridge Test P...
• Sandbox

Webhooks

My App

- Preview
- Settings
- Customization

Items & Users

- Connections status
- Performance monitoring
- Connect conversion rate

Payments

- Payments list
- Payments analytics
- Payment links
- Payment links analytics

Webhooks

You do not have webhooks yet

You can use our webhooks to be notified about specific events like new data on an account, an update on an item, etc. [Access webhooks documentation](#)

Add a webhook

Finally, within your WebHook, make sure there is good communication between your Bridge account and your Magento environment by sending a test:

Disable Send a test Delete

Edit webhook

Callback URL

Ex: <https://api.yourdomain.com/your-endpoint>

Name (optional)

An optional webhook name or description

Auto-generated secret

***** Regenerate secret

Use this to secure your webhook. If you regenerate it, the previous secret will be valid for 24 hours. [See documentation](#)

This test must trigger your Magento WebHook, which must be validated in the 'Status' tab

Status

- ✓ CURL - version v7.74.0 OpenSSL/1.1.1n
- ✓ SSL & TLS1.2 - Enabled
- ✓ Connected to bridge API - 181 banks
- ✗ Production enviroment
- ✓ WebHook contacted - Ok

WebHook configuration

<https://bridgemagento.magento.202-dev.co...> 

This URL is needed to connect Bridge to your website. It must be entered in the WebHook section of your Bridge account for the module to work properly.

5. Use of the module

Once the configuration is complete, the payment method is active on your payment tunnel.

Place an order on the test API in order to validate the entire payment flow and to validate that the order goes back to the back office of your Magento.

Once this verification is done, all you have to do is **select the production environment**.

Environment [store view]	Production	▼
-----------------------------	------------	---

This option defines in which environment your module is configured

Client ID (Production) [store view]	16e61de8b8de4c2aa5a1a8806479052b
--	----------------------------------

This information is located in your Bridge dashboard: 'Settings' > 'General settings'

Client Secret (Production) [store view]	*****
--	-------

This information is located in your Bridge dashboard: 'Settings' > 'General settings'

Status

- ✓ CURL - version v7.74.0 OpenSSL/1.1.1n
- ✓ SSL & TLS1.2 - Enabled
- ✓ Connected to bridge API - 181 banks
- ✓ Production environment
- ✓ WebHook contacted

The customer of your store can then choose their bank and make a payment by bank transfer.

Payment Method

Check / Money order

Bridge 

Choose my bank

Back

	BNP Net Entreprises (Lecteur sans fil)
	BNP Net Entreprises (Navigation Sécurisée)
	BNP Paribas Particulier
	BNP Paribas Privée

My billing and shipping address are the same

Place Order