



Cardlink Checkout Payment Gateway v.1.3.0
for Magento 2.x

Date: 12/05/2026

Cardlink Checkout Payment Gateway for Magento 2.x

Contributors: cardlink

Tags: payments, payment-gateway

Requires at least: 2.2.6

Tested up to: 2.4.78

Requires PHP: 7.x - 8.x

License: GPLv2 or later

License URI: <http://www.gnu.org/licenses/gpl-2.0.html>

Changelog

- **1.3.0**
 - Support for background confirmations.
 - Support secondary transactions (capture, refund, cancel) through VPOS XML API calls.
- **1.2.5**
 - Fixes for PHP 7.x support.
- **1.2.4**
 - Remove requirement for IRIS customer code configuration setting. (If you already use IRIS with your own seller id, DO NOT install the new version!)
- **1.2.3**
 - Add Cardlink and Worldline as IRIS business partners. (If you already use IRIS with your own seller id, DO NOT install the new version!)
 - Various fixes for IRIS payments.
- **1.2.2**
 - Fix minor IRIS redirection issues.
 - Enforce Terms and Conditions acceptance requirement on checkout.
- **1.2.1**
 - Minor fixes in the cron task and related admin field.
 - Limit added to that field with a valid range of 10-60 minutes. Leave empty to disable feature.
 - Updated README.
- **1.2.0**
 - Added configurable order creation and stock management (option to create order before or after successful payment)
 - Enhanced automatic order cancellation system (option to configure expiration time)
 - Updated README.
- **1.1.2**
 - Require merchant information when activating payment methods.
- **1.1.1**
 - Minor fixes.
 - Updated README.
 - Added check for payment method code in OrderPaymentSaveBefore observer.
- **1.1.0**
 - Added support for IRIS payments.
- **1.0.5**
 - Fixed guest checkout bug.

- **1.0.4**
 - Minor bug fixes.
- **1.0.3**
 - Order creation only happens after successful payment.
 - Fixed bug with credit memo creation.
 - Fixed bug with reorder in admin.
- **1.0.2**
 - Fixed a bug that failed checkout with PayPal Standard.
 - Changed auto cancel order period to 60 minutes.
- **1.0.1**
 - Only send order confirmation email to customer when the order has been successfully paid for.
 - Auto cancel order after 30 minutes in pending payment state.
- **1.0.0**
 - Initial release

Support tickets

In case that you face any technical issue during the installation process, you can contact the Cardlink e-commerce team at ecommerce_support@cardlink.gr.

Description

Cardlink Payment Gateway allows you to accept payment through various schemes such as Visa, Mastercard, Maestro, American Express, Diners, Discover cards on your website, with or without variable installments.

This module aims to offer new payment solutions to Cardlink merchants for their Magento 2.x online store without having web development knowledge. However, for the initial module installation some technical knowledge will be required.

Merchants with e-shops (redirect cases only) will be able to integrate the Cardlink Payment Gateway into their checkout page using the CSS layout that they want. They can also choose between the redirect and IFRAME options for the payment environment. Once the payment is made, the customer returns to the online store and the order is updated.

Once you have completed the requested tests and any changes to your website, you can activate your account and start accepting payments.

Features

1. A dropdown option for instance between Worldline, Nexi and Cardlink.
2. Option to enable test environment (sandbox). All transactions will be re-directed to the endpoint that represents the production environment by default. The endpoint will be different depending on which acquirer has been chosen from instance dropdown option.
3. Ability to define the maximum number of installments regardless of the total order amount.
4. Ability to define the ranges of the total order amounts and the maximum number of installments for each range.
5. Option for pre-authorization or sale transactions.
6. Option for a user tokenization service. The card token will be stored at the merchant's e-shop database and will be used by customers to auto-complete future payments.

7. In-store checkout option: the merchant can configure the payment process to take place inside a pop up with IFRAME to mask the redirection process from the customers.
8. A text field for providing the absolute or relative (to Cardlink Payment Gateway location on server) URL of custom CSS stylesheet, to apply custom CSS styles in the payment page.
9. Translation ready for Greek & English languages.
10. Configurable automatic cancellation of pending orders.
11. Only send the order confirmation email to the customer after a successful payment.
12. Support for IRIS payments for Worldline, Nexi and Cardlink acquirers.
13. The IFRAME feature is not supported for IRIS payments.
14. Configurable order creation (before or after successful payment).

Background Confirmation Endpoint (Webhook)

The module provides a background confirmation endpoint for server-to-server (S2S) notifications from the payment gateway. This endpoint processes the same payment confirmation data as the regular Response controller but returns a JSON response instead of redirecting the user.

Configure this URL in the Cardlink VPOS back-office as the **background confirmation URL** for your merchant account:

```
https://<your-magento-store>/cardlink_checkout/payment/backgroundconfirmation
```

Endpoint Details

Property	Value
URL	<code>https://<your-magento-store>/cardlink_checkout/payment/backgroundconfirmation</code>
Relative Path	<code>/cardlink_checkout/payment/backgroundconfirmation</code>
HTTP Method	POST
Content-Type	<code>application/json</code> (response)

Security

The endpoint does **not** require HTTP authentication credentials, but every request is validated before processing:

- The `User-Agent` header must identify the Cardlink gateway HTTP client.
- The request must arrive over HTTPS in production (non-HTTPS is only allowed in sandbox/test mode).
- All required fields (`Status`, `OrderId`, `Digest`, `MerchantId`) must be present.
- The HMAC-SHA256 cryptographic signature (`Digest`) is verified against the shared secret configured for the payment method (standard card, IRIS).
- The `MerchantId` in the request must match the merchant ID in the module configuration.

Requests that fail any of these checks are rejected with HTTP 400.

JSON Response Format

```
{
  "success": true,
```

```

    "status": "ok",
    "message": "Payment processed successfully",
    "order_id": 123,
    "order_increment_id": "000000001",
    "order_state": "processing",
    "order_status": "processing"
}

```

Response Status Values

Status	Description
ok	Payment was successfully processed
processed	Failed/canceled payment was processed
already_processed	Order was already processed (idempotent)
error	Invalid signature or unknown payment status

HTTP Response Codes

Code	Description
200	Request processed successfully
400	Invalid request (bad signature, missing fields, or unknown status)
500	Internal server error

Installation

You need to manually upload the contents of the .zip file of the module's latest version to your server's web root folder that your Magento store is installed.

You will first need to extract the file's contents to a temporary folder. Then, go to the `Cardlink\Checkout\Controller\Payment` folder.

If you are using previous Magento version 2.2.x, replace the `Response.php` file with the contents of file `Response-v2.2.php`.

A message will be displayed during Magento 2 setup commands if this file has not been replaced, so that the administrator is aware.

Depending on your hosting provider, you will probably have to be familiar with the process of transferring files using an FTP or SFTP client. If no FTP/SFTP access is provided, use your hosting provider's administration panel to upload the folders to the folder of your Magento installation.

Using an SSH console, go to your web root folder (or the folder your Magento store is installed) and execute the following commands to enable the extension and execute its installation scripts.

```

# bin/magento module:enable Cardlink_Checkout
# bin/magento setup:upgrade
# bin/magento setup:di:compile

```

Required Hosting Settings

For security reasons, Web browsers will not send target domain cookies when the referrer website is on another domain and data are POSTed unless the `SameSite` option of these cookies is set to the value `None`. If you fail to properly configure the required hosting settings, customers returning from the payment gateway will be automatically logged out from their

accounts. The following configuration instructions will manipulate all cookies set by your store to allow customer sessions to persist after returning from the payment gateway.

Apache Web Server

For hosting solutions running the Apache web server software, you will need to add the following lines to your web site's root `.htaccess` file. Make sure the `mod_headers` Apache module is installed and active.

```
<IfModule mod_headers.c>
Header always edit Set-Cookie ^(.*)$ $1;SameSite=None;Secure
</IfModule>
```

Nginx Web Server

If your hosting provider uses the Nginx web server instead, you will need to add/edit the following lines of code to your virtual host's configuration file.

```
location / {
    proxy_cookie_path / "/; SameSite=None; Secure";
    ...
}
```

Plesk Hosting Control Panel

If you are using Plesk and nginx in proxy mode, under `Apache & nginx Setting for ... > Additional nginx directives` add only the following line:

```
proxy_cookie_path / "/; SameSite=None; Secure";
```

If you are only using Apache, add the following configuration lines in the `Additional Apache directives` section on the same page. By default, Plesk has the Apache `mod_headers` module installed and active; however, verify that this is the case for your Plesk installation.

```
<IfModule mod_headers.c>
Header always edit Set-Cookie ^(.*)$ $1;SameSite=None;Secure
</IfModule>
```

Custom or Unknown Hosting

If you do not have full control of your hosting services, or not very confident with such changes, you can try installing the [Veriteworks CookieFix](<https://github.com/Veriteworks/CookieFix>) extension. Follow the instructions on the extension's GitHub page and set the SameSite setting to None. If you are still unsure or unfamiliar with the actions described above, please ask a trained IT person or contact your hosting provider to do them for you.

Screenshots

1. The Cardlink Payment Gateway settings screen used to configure the main Cardlink gateway (`Stores > Configuration > Sales > Payment Methods`).

Configuration


Store View: Default Config

Save Config

- GENERAL
- CATALOG
- CUSTOMERS
- SALES
- SALES Emails
- PDF Print-outs
- Tax
- Checkout
- Shipping Settings
- Multishipping Settings
- Shipping Methods
- Payment Methods
- Fraud Protection
- SERVICES
- ADVANCED

Merchant Location

RECOMMENDED SOLUTIONS:



PayPal Express Checkout
Add another payment method to your existing solution or as a stand-alone option. [Configure](#)

[Check out with PayPal](#)

OTHER PAYPAL PAYMENT SOLUTIONS: ▼

OTHER PAYMENT METHODS: ▲

Cardlink Payment Gateway Checkout

Enabled (store view) Yes

Title (store view)
The title of the payment method to be displayed during the checkout.

Description (store view)
A short description of the payment method to be displayed during the checkout.

New Order Status (store view) Use system value

Payment from Applicable Countries (store view) Use system value

Payment from Specific Countries (store view)

Business Partner (store view)
Identify the business partner that will handle payment transactions as agreed with Cardlink.

Merchant ID (store view)
The merchant ID provided by Cardlink.

Shared Secret (store view)
The shared secret code provided by Cardlink.

Transactions Environment (store view)
Identify the working environment for payment transactions.

Payment Action (store view)
Identify the type of transaction to perform. By selecting the Authorize option, you will need to manually capture the order amount on Cardlink's merchant dashboard.

Accept Installments (store view)
Enable installment payments and define the maximum number of installments.

Maximum Installments (store view)
The maximum number of installments available for all orders.

Allow Tokenization (store view) Yes
Enable customers to select whether they want to secure store their payment cards for future checkouts.

Checkout without Leaving Your Store (store view) Yes
Perform the payment flow without having the customers leave your website for Cardlink's payment gateway. You will need to have a valid SSL certificate properly configured on your domain.

Force Store Language on Cardlink's Payment Gateway (store view) Yes
Instruct Cardlink's Payment Gateway to use the language of the store that the order gets placed.

Display Cardlink Logo (store view) Yes
Display the Cardlink logo next to the payment method title.

CSS URL (store view)
Full URL of custom CSS stylesheet, to be used to display payment page styles.

Log Debug Information (store view) No
Store debug log information in file cardlink.log. Use only for a limited time frame and delete the log file afterwards.


Sort Order (store view)

- Check / Money Order
- Bank Transfer Payment
- Cash On Delivery Payment
- Zero Subtotal Checkout
- Purchase Order
- Authorize.net Direct Post

Copyright © 2022 Magento Commerce Inc. All rights reserved.

Magento ver. 2.2.6
[Report an Issue](#)


- This is the front end of the Cardlink Payment Gateway plugin, located on the checkout page.

🔍 🛒 1

Estimated Total 🛒 1
€39.00

Payment Method:

Pay through Cardlink



My billing and shipping address are the same

TEST TEST
TEST
TEST, 151 26
Greece
+301234567890

Number of Installments *
6

Stored Cards *
 VISA xxxx-0010
12/2032
 New Card
 Securely store card

Place Order

[Apply Discount Code](#) ▾

Subscribe

- About us
- Customer Service
- Privacy and Cookie Policy
- Search Terms
- Advanced Search
- Contact Us

Copyright © 2013-present Magento, Inc. All rights reserved.

- To set up IRIS payments, you will need to have the Merchant ID and Shared Secret specifically issued for use with IRIS. Other settings are similar to the ones for Card Payments.

Cardlink CheckOut - Pay via IRIS

Display settings for the 'Pay via Card' method are shared with this method.

Enabled [store view] Yes

New Order Status [website] Pending Use system value

Payment from Applicable Countries [website] All Allowed Countries Use system value

Payment from Specific Countries [store view]
 Afghanistan
 Åland Islands
 Albania
 Algeria
 American Samoa
 Andorra
 Angola
 Anguilla
 Antarctica
 Antigua & Barbuda

Sort Order [store view] 2

Title [store view] Pay via IRIS
 The title of the payment method to be displayed during the checkout.

Description [store view] Pay via IRIS.
 A short description of the payment method to be displayed during the checkout.

Business Partner [store view] Nexi
 Identify the business partner that will handle payment transactions as agreed with Cardlink.

Merchant ID [store view] 70000186
 The merchant ID provided by Cardlink.

Shared Secret [store view]
 The shared secret code provided by Cardlink.

Transactions Environment [store view] Sandbox
 Identify the working environment for payment transactions.

DIAS Code [store view] 90002
 The merchant's code on the DIAS network.

Display IRIS Logo [store view] Yes
 Display the IRIS logo next to the payment method title.

CSS URL [store view]
 Full URL of custom CSS stylesheet, to be used to display payment page styles.

4. To create orders (and thus hold stock) before successful payment, set the option to Yes. Setting the option to No will create the order after successful payment. Forgotten or pending order payment requests are automatically cancelled after the expiration time.

Create Order and hold stock 
[store view]

Yes to create order and hold stock, No to create order only after successful payment

Cancel order after
[store view]

New order expires after N minutes. Valid range: 10 to 60 minutes. Leave empty to disable.