



Verifone Payment gateway for Magento2 eCommerce platform

Table of contents

1. Introduction
2. Installation and Configuration
3. Payment service key handling
4. Server configuration
5. Late shop payment success page URL
6. Troubleshooting
7. Order states and inventory handling
8. Refund feature
9. Configuration summary

1. Introduction

This document is the Verifone payment module installation and configuration guide.

Verifone payment module offers Magento2 integration towards the Verifone payment service.

2. Installation and Configuration

1. Navigate to your Magento project directory and update your composer.json file:
composer require verifone/module-payment
2. Wait for Composer to finish updating your project dependencies and make sure there aren't any errors.
3. To verify that the extension installed properly, run the following command: **bin/magento module:status**
4. Enable the extension and clear static view files: **bin/magento module:enable Verifone_Payment --clear-static-content**
5. Register the extension: **bin/magento setup:upgrade**
6. Recompile your Magento project: **bin/magento setup:di:compile**
7. Verify that the extension is enabled: **bin/magento module:status**
8. You may need to clear the cache if you encounter errors when loading the storefront in a browser: **bin/magento cache:flush**
9. Configure the extension in Admin: **Stores > Configuration > Sales > Payment Methods > Verifone Payment**

10. Enter values for Configuration (on screen values for test environment):
11. Save configuration.
12. Click **Get Available Payment Methods** button to get all payment methods. By using **Available payment method** you may show payment methods as separated Payment Methods and customer will be redirected directly to the bank or card data form. If you

wish to use Verifone Gateway then select **Verifone Payment - All in One**. Tip: use CTRL + [click] to select multiple options.

13. For check stored configuration, please click on **Display configuration summary**. More information in section 10.

Verifone Payment

Display configuration summary

Display configuration summary

Enabled (website) Yes

Title (store view) Verifone Payment

Sort Order (website) 1

Module on live mode (website) Test

Select if extension is on live mode (Production) or test mode (Test)

Verifone Payment merchant agreement code (website) EC9988776655

Required. Verifone Payment agreement code

Verifone Payment merchant agreement code (Test) (website) demo-merchant-agreement

Required. Verifone Payment agreement code

Payment service key handling (website) Automatic (Simple)

Generate live keys

When you generate live keys, you will need to upload the new public key to Verifone Payment portal

Generate test keys

Uses preset keys by default, only needed if using custom test agreements

Skip confirmation page (website) Yes

Return directly to shop after payment

Style code (website) test-style

Use of custom payment page template needs first to be uploaded and to be approved by Verifone Payment

Disable rsa blinding (website) Yes

Define CRYPT_RSA_DISABLE_BLINDING as true in case of custom PHP build or PHP7 (experimental)

Basket Item Sending (website) Send for all payment methods

Combine Invoice Basket Items (website) Yes

Currently only available for invoice payment methods

Get Available Payment Methods

Select payment methods and set groups (website)

Group Position	Group name	Payments	
1	Banks	<ul style="list-style-type: none"> OP_POHJOLA_VERKKOMAKS NORDEA_E_PAYMENT SAMPO_WEB_PAYMENT HANDELSBANKEN_E_PAYME ALANDSBANKEN_E_PAYMEN INVOICE_COLLECTOR SVEA_WEBPAY_INVOICE SAASTOPANKIN_VERKKOMA POP_PANKIN_VERKKOMAKSI OMA_SAASTOPANKIN_VERKI 	<p>Edit order</p> <p>Delete</p>
Group Description			

Add new group

Card payments group (website)

Group Position	Group name	Payments	
5	Cards	<ul style="list-style-type: none"> VISA MASTER_CARD AMEX DINERS 	<p>Edit order</p>
Card Group Description			

Add new group

Allow to save Credit Cards (website) Yes

Use external customer id. (website) No

If no, then telephone from billing address will be use as identificator. If yes, please select field from list below.

Remember me info (website)

Optional. Note In checkout after Remember payment method checkbox - for cards only

Save masked PAN number (website) Yes

Payment from Applicable Countries (website) All Allowed Countries

Payment from Specific Countries (website)

- Angola
- Anguilla
- Antarctica
- Antigua and Barbuda
- Argentina
- Armenia
- Aruba
- Australia
- Austria
- Azerbaijan

Minimum Order Total (website) 0

Maximum Order Total (website)

Leave empty to disable limit

New Payment Order Status (website) Pending Verifone Payment

Completed Payment Order Status (website) Processing

Configuration setting	Description
Display configuration summary	Possibility to display summary for shop configuration. Available information about which merchant agreement or keys is used. Before check configuration for test/live you need first save the configuration. The summary is available per Website Scope, so the first website should be selected from the scope selector. If the website is not selected, then Default values are visible. See section 9 for more information.
Title	Name of the payment module. This name is visible to the customer as the name of the payment.
Enabled/Disabled	The payment method can be enabled/disabled with this selection.
Sort order	Display order of the payment in customer's payment selection.
Environment (Test/Production)	Select the environment for the payment module
Verifone Payment production merchant agreement code	Unique seller identification obtained from Verifone.
Verifone Payment test merchant agreement code	Seller identification used for the test.
Payment service key handling	Select which key handling mode you will use. See section 3 for more information.
Generate new keys	Functionality for creating a new key pair for the shop. See section 3 for more information.
Pay page URL 1,2,3	Payment system endpoint address (payment node). For redundancy, it is possible to have up to three separate endpoints. The payment module will choose the endpoint used randomly.
Delayed success url	This URL should be provided into Verifone. The URL is used for fetch payment transaction when payment takes more time.

Check payment node availability	Before payment, the payment node is polled in order to check its availability. In case a certain payment node is not available, another one is checked. In case no payment node is available an error is raised. The availability check can be bypassed with this configuration setting.
Directory for store keys	Required. A path to the directory for generated files Important: Do not place keys in a folder accessible by the WWW-server.
Shop private key filename	The filename for the shop key file. Important: Do not place keys in a folder accessible by the WWW-server.
Shop private key filename (Test)	The filename for the shop key file used for the test. Important: Do not place keys in a folder accessible by the WWW-server.
Skip confirmation page	In the Verifone payment system, it is possible to skip the last page and direct user directly to the shop. With this selection, the functionality can be turned on/off.
Style code	Code identifying the style sheet used in payment page or an empty string if the default style sheet is used. Use of custom payment page template needs first to be uploaded and to be approved by Verifone.
Disable rsa blinding	Define CRYPT_RSA_DISABLE_BLINDING as true in case of custom PHP build or PHP7 (experimental)
Basket Item Sending	Select for which type of order the items should be sent. When you select the option for send items, then the product list is provided for Verifone payment.
Combine Invoice Basket Items	If yes, then all products are sent to Verifone as one product with name Tilau %ORDERNUMBER%
Get Available Payment Methods	Option to get payment methods from Verifone gate and redirect user directly to the payment page, dodging portal step

Available payment methods	Select visible payment methods. To fill the list click „ Get Available Payment Methods ” button.
Allow to save Credit Cards	Allow customer to store credit cards in Verifone for future payments.
Use external customer id	If you want to use another field to match a user in Verifone service, please set field name. If no, then telephone from billing address will be used as identification. If yes, please select a field from the list below.
Field for external customer id	Selected field will be used to identify customer
Save masked PAN number	Save masked pan number to order
Remember me info	Information visible when Allow to save Credit Cards is set to Yes
Payment from Applicable Countries	Select if payment should be available for all countries, or should be restricted
Payment from Specific Countries	Select available countries
Minimum Order Total	Leave empty if you do not want to limit order minimum
Maximum Order Total	Leave empty if you do not want to limit order maximum
New Payment Order Status	Set status for order when payment is pending
Completed Payment Order Status	Set status for order when payment is done and received

3. Payment service key handling

For testing payments, the module contains test keys, and uses it, when the simple mode is configured.

Automatic (Simple)

- When everything is configured, please run **Generate live keys** (for production) or **Generate test key** (for development/test). Confirmation for this action is required.
- Generation a new test key is required only when custom merchant agreement code is configured in field **Verifone Payment test merchant agreement code**
- After creating a new key, remember to copy the public key to payment operator configuration settings. The public key is available in the **Configuration summary**.
- **Generate new keys** to override old keys. Please don't click this link if you are not sure.

Manual (Advanced)

- This configuration could be used for keys created in the old way.
- Please configure the directory path where keys are stored.
- Please configure file name for production and/or test environment.

By clicking on Display configuration summary you can check which keys are being used. If you encounter any problems, please see section 6. Troubleshooting

4. Server configuration

Required configuration for the http-server:

- SSL-support installed and enabled (e.g. OpenSSL for Apache web server)
- allow_url_fopen in PHP configuration must be set on.

Cron is required for check payment status for pending orders. For information about configuring it, please check Magento documentation:

<https://devdocs.magento.com/guides/v2.3/config-guide/cli/config-cli-subcommands-cron.html>

5. Late/delayed shop payment success page URL

The late shop payment success page URL is:

https://<yourdomain>/verifone_payment/payment/successDelayed/ or
http://<yourdomain>/verifone_payment/payment/successDelayed/

depending on whether SSL is used or not.

Verifone may request this information for production use web shops.

Full delayed url is available in the module information view, after a click on “**Display configuration summary**”

6. Troubleshooting

Problem	Resolution
Verifone payment is not visible in payment selection	Check configuration values. Check that the module is enabled.
Error: Payment system not available.	The payment system availability check has failed. Check Pay Page URL parameters.
Error: Problem with generating new keys.	Check directory configuration. A directory must be accessible and writable.
Error: Problem with directory configuration.	Check directory configuration. A directory must be accessible and writable.
Error: Key file for provided merchant agreement code does not exist	This problem appears, when is set merchant agreement code or custom key file name, but the key does not exist. Check configuration, and if required, then create new keys. Remember about upload into payment service.
Info: Default test key file uses	The plugin uses default keys, provided with the module. It is using for the default configuration.
Error: Key file could not be read.	The payment module has not been able to read the key file. Possible causes are: <ul style="list-style-type: none"> • A key file is not stored to the server • Key file path or filename is wrong • Key file does not have reading rights. Check that the key file exists in the server, check that the path and file name is correct and check file permissions.
Error: Payment verification failed	The payment verification has failed. Check that the key file is correct.
Error: Configuration error: allow_url_fopen must be enabled	Change allow_url_fopen true in PHP

7. Order states and inventory handling

Order state before payment has been accepted will be getting from **New Payment Order Status**.

For successfully paid orders, the order state will be gotten from **Completed Payment Order Status**.

For cancelled orders, the state will be set as **Canceled**. Order cancellation will be done, when the customer cancels payment in the Verifone payment system and returns to the shop, or payment is not finalized.

Note that the order is created when the customer begins the payment process. The sole existence of an order does not indicate, that the order is actually ordered and paid. Only orders, which are on the state will be gotten from **Completed Payment Order Status**, are to be delivered.

When stock management is enabled, the ordered quantity is reserved from the stock at the time of order creation. For orders, where the payment process is never completed nor

canceled, the order with associated stock reservation remains indefinitely at **New Payment Order Status** configuration. Manually cancelling the orders will return items to the stock.

000000081	Main Website Main Website Store Default Store View	May 08, 2017, 16:05:00 PM	simon@lamia.fi Lamia Testing	simon@lamia.fi Lamia Testing	€27.00	€27.00	Pending Verifone	View	visa
000000080	Main Website Main Website Store Default Store View	Apr 12, 2017, 16:04:00 PM	simon@lamia.fi Lamia Testing	simon@lamia.fi Lamia Testing	€27.00	€27.00	Processing	View	visa
000000079	Main Website Main Website Store Default Store View	Apr 04, 2017, 9:04:00 AM	simon@lamia.fi Lamia Testing	simon@lamia.fi Lamia Testing	€27.00	€27.00	Canceled	View	visa

At the example above order:

- #81 is waiting for payment.
- #80 has been paid and can be delivered,
- #79 has been cancelled.

8. Refund feature

To refund orders, you need to go to the order details page, select invoice from invoices tab by clicking on the **View** link. After the invoice page has been loaded, you need to click on the **Credit Memo** button.

Next up, fill the credit memo form like for any other refund, and click the **Refund** button to return sufficient amount to the customer.

Refund Totals

Subtotal	€22.00
Refund Shipping	<input type="text" value="5"/>
Adjustment Refund	<input type="text" value="0"/>
Adjustment Fee	<input type="text" value="0"/>
Grand Total	€27.00

Append Comments

Email Copy of Credit Memo

Refund Offline

Refund

9. Configuration summary

Below are available screenshots with example configuration summary for different states.

Configuration summary is available after click **Display configuration summary** button.

Display configuration summary

Display configuration summary

Enabled
[website]

Yes

Title
[store view]

Verifone Payment

1. Default test configuration - use for testing with default keys.

Configuration summary - Main Website

Mode	Test
Verifone Payment merchant agreement code	demo-merchant-agreement Default test merchant agreement uses
Delayed success url	http://magento2.dev/verifone_payment/payment/successDelayed/ This is the url that you need to copy to payment provider settings in their portal.
Key handling mode	Automatic (Simple)
Path and filename of Verifone Payment public key file	<code>/var/docker_storage/workspace/magento2/app/code/Verifone/Payment/keys/verifone-e-commerce-test-public-key.pem</code> Key file is available
Path and filename of shop private key file	Key file stored in database Default key file is used

2. Custom test configuration - use for testing with custom merchant agreement code and keys

Configuration summary - Main Website

Mode	Test
Verifone Payment merchant agreement code	test-merchant-agreement
Delayed success url	http://magento2.dev/verifone_payment/payment/successDelayed/ This is the url that you need to copy to payment provider settings in their portal.
Key handling mode	Manual (Advanced)
Path and filename of Verifone Payment public key file	<code>/var/docker_storage/workspace/magento2/app/code/Verifone/Payment/keys/verifone-e-commerce-test-public-key.pem</code> Key file is available
Directory for store keys	<code>/var/docker_storage/workspace/keys</code> Directory configured properly
Path and filename of shop private key file	<code>/var/docker_storage/workspace/keys/test-merchant-private.pem</code> Key file is available

3. Production configuration

Configuration summary - Main Website

Mode	Production
Verifone Payment merchant agreement code	EC9988776655
Delayed success url	http://magento2.dev/verifone_payment/payment/successDelayed/ This is the url that you need to copy to payment provider settings in their portal.
Key handling mode	Automatic (Simple)
Path and filename of Verifone Payment public key file	<code>/var/docker_storage/workspace/magento2/app/code/Verifone/Payment/keys/verifone-e-commerce-live-public-key.pem</code> Key file is available
Path and filename of shop private key file	Key file stored in database Key file is available
Public key file	<pre>-----BEGIN PUBLIC KEY----- MIGfMA0GCsqGSib3DQEBAQUAA4GNADCBiQKBgQC5BREeg4GNhrLSIc+ZLAQzYwSu 5AyAqKjxz0sJE5akGTVowdKFhEJ8BCKFY7Z8iFNog+sYBLsiIRxiNwVx6upG6Q0t fwc0EWGBrPgk2In+nHG64dKbn/BXEZ1KBriG02U9kxH1iuARged4jMI95aQ+lmb5 uvvfQaZK8H5xLvk/ywIDAQAB -----END PUBLIC KEY-----</pre> Please, copy this key to payment operator configuration settings, otherwise, the payment will be broken.