**Klarna Payments for SFRA**

Version 18.1.0



Table of Contents

[1. Overview 1-3](#_Toc531019244)

[1.1 Authorize payments and place orders with Klarna using SFRA single-page checkout 1-3](#_Toc531019245)

[1.2 Ability to use Klarna Payments SFRA for US and international sites 1-3](#_Toc531019246)

[1.3 Multitude of payment options the customer can choose from 1-3](#_Toc531019247)

[1.4 Handling notifications 1-4](#_Toc531019248)

[1.5 Creation of VCN Settlements for each order 1-5](#_Toc531019249)

[1.6 Widget customizations 1-5](#_Toc531019250)

[2. System Extensions 2-7](#_Toc531019251)

[2.1 System object extensions 2-7](#_Toc531019252)

[2.1.1 Order 2-7](#_Toc531019253)

[2.1.2 PaymentTransaction 2-7](#_Toc531019254)

[2.1.3 OrderPaymentInstrument 2-8](#_Toc531019255)

[2.2 Custom objects 2-8](#_Toc531019256)

[2.2.1 KlarnaCountries 2-8](#_Toc531019257)

[2.3 Site preferences 2-9](#_Toc531019258)

[2.3.1 Klarna\_Payments group 2-9](#_Toc531019259)

[2.4 Services 2-10](#_Toc531019260)

[3. Installation 3-11](#_Toc531019261)

[3.1 Verify Package Contents 3-11](#_Toc531019262)

[3.2 Cartridge upload 3-11](#_Toc531019263)

[3.3 Metadata import 3-11](#_Toc531019264)

[3.4 Service configuration 3-12](#_Toc531019265)

[3.5 Add cartridge to cartridge path 3-12](#_Toc531019266)

[3.6 (optional) Enable VCN 3-13](#_Toc531019267)

[4. Integration with other payment cartridges 4-14](#_Toc531019268)

[5. Uninstallation 5-15](#_Toc531019269)

[6. Release History 6-16](#_Toc531019270)

[7. Additional Information 7-17](#_Toc531019271)

[7.1 Klarna API Information 7-17](#_Toc531019272)

[7.1.1 Live environment 7-17](#_Toc531019273)

[7.1.2 Testing environment 7-17](#_Toc531019274)

[7.2 Generate SSL Keys for VCN Settlements 7-17](#_Toc531019275)

# Overview

**Klarna Payments SFRA** is a cartridge built for SFRA-based sites which allows customers to select payment options and place orders with Klarna using the SFRA single-page checkout.

Here’s a list of key features:

* Authorize payments and place orders with Klarna using SFRA single-page checkout.
* Ability to use Klarna Payments for US and international sites built upon the SFRA.
* Multitude of payment options (pay later, slice, etc.) the customer can choose from.
* Creation of VCN settlements for each order.
* Handling of pending orders acceptance/rejection due to anti-fraud checks via notifications.
* Site managers can customize the Klarna Payments widget styling, so it matches the style guide of their website(s).

## Authorize payments and place orders with Klarna using SFRA single-page checkout

Successful payment authorizations (Klarna Payments status APPROVED) lead to a 'Paid' order payment status and ‘Ready for Export’ export status

Refused and pending payment authorizations (Klarna Payments statuses REJECTED and PENDING) lead to a 'Not Paid' order payment status, and ‘Not Exported’ export status

Cancels and refunds lead to a 'Not Paid' order payment status (even if the status was 'Paid' before), and ‘Not Exported’ export status

The cartridge makes use of the Klarna Payments JSON REST API and a JavaScript SDK.

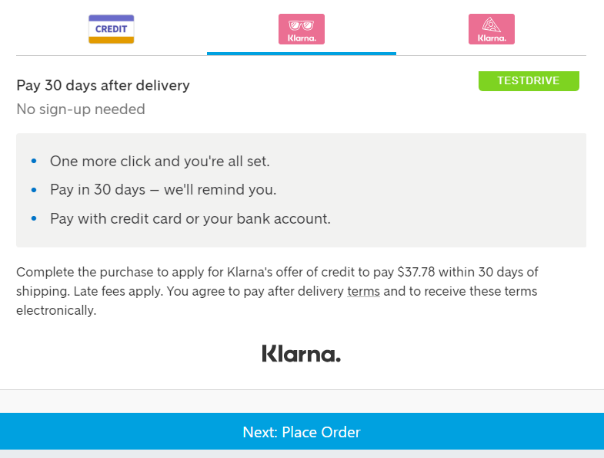
## Ability to use Klarna Payments SFRA for US and international sites

Klarna Payments SFRA can be configured independently on each site and each locale.

## Multitude of payment options the customer can choose from

On the checkout billing page, Klarna payment options are going to be dynamically loaded based on submitted customer information prior to that.

The screen below shows two payment options – pay later and slice:



The following guide describes the integration process of this cartridge; how to install, configure and use the cartridge.

## Handling notifications

Notifications are received when an order has been changed in the Klarna merchant portal. For example, when an order is placed by a customer, Klarna may hold the order for further fraud checks. Thus, the order will be left in status CREATED.

After a while, Klarna may send a notification to SFCC with one of several event types: ACCEPTED, REJECTED, STOPPED. No matter of the event, the order’s payment transaction is updated (see *kpFraudStatus*). This can be seen in BM on the order details Payment tab as below:



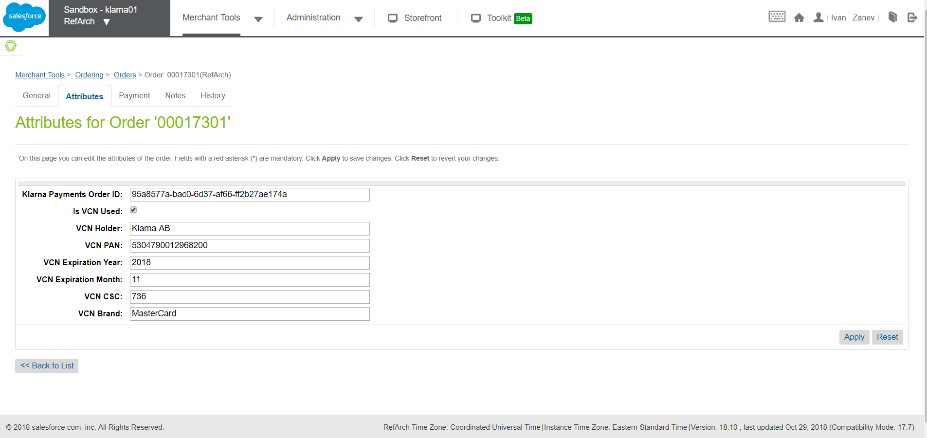
If the order is ACCEPTED upon notification, the order will be placed in SFCC (order status changes to OPEN). If VCN is enabled and configured properly, a VCN settlement is created, authorized and the information is saved within the order attributes (can be seen in the Attributes tab).

If the order is REJECTED or STOPPED upon notification, the order is failed in SFCC.

## Creation of VCN Settlements for each order

The ability to create VCN settlements for each order is a preference within the Business Manager.

An order placed with the VCN settlement turned on is going to have attributes related to the VCN settlement as show below:



It is up to the merchant to use the credit card details as described above to authorize the virtual credit card.

## Widget customizations

The merchant will need a configured Klarna Payments account.

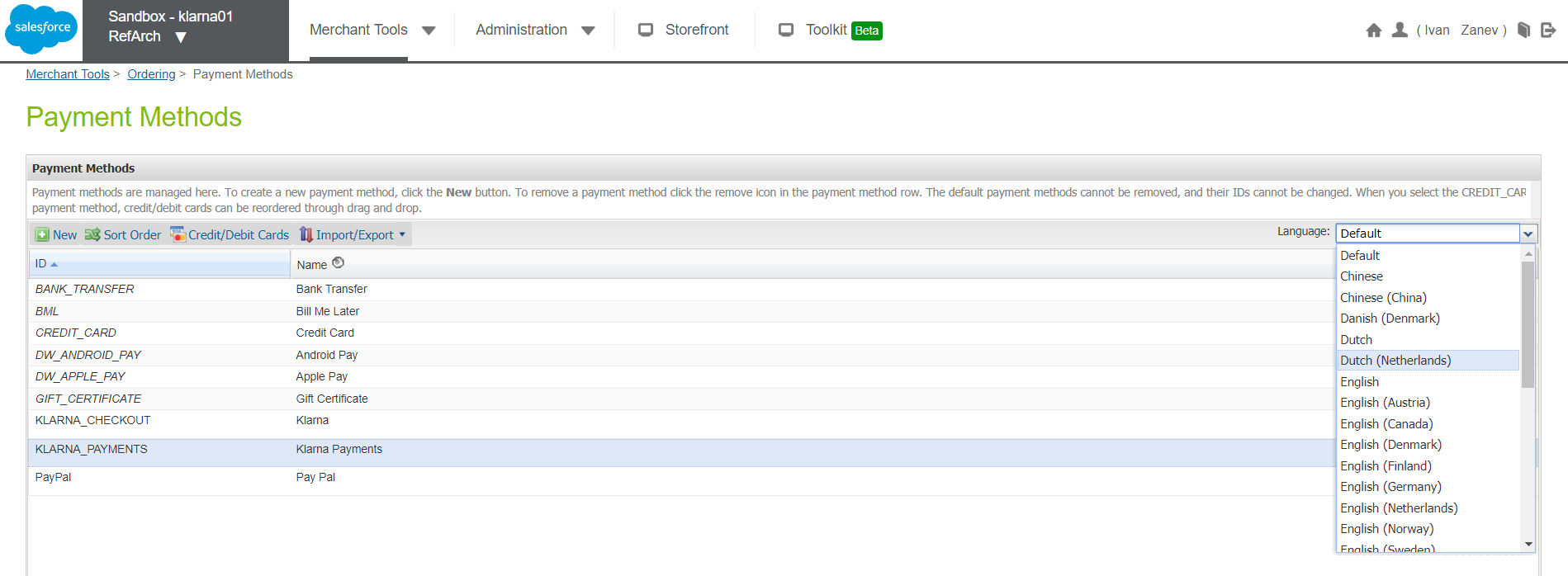
The merchant can style the Klarna Payments widget (skin), to make it matching the look and feel of their store. The list with the graphic elements that can be customized, through site preferences is as follows:

* "color\_details" (site preference kpColorDetails): "#C0FFEE"
* "color\_button" (site preference kpColorButton): "#C0FFEE"
* "color\_button\_text" (site preference kpColorButtonText): "#C0FFEE"
* "color\_checkbox" (site preference kpColorCheckbox): "#C0FFEE"
* "color\_checkbox\_checkmark" (site preference kpCheckboxCheckmark): "#C0FFEE"
* "color\_header"(site preference kpColorHeader): "#C0FFEE"
* "color\_link"(site preference kpColorLink): "#C0FFEE"
* "color\_border"(site preference kpColorBorder): "#C0FFEE"
* "color\_border\_selected"(site preference kpBorderSelected): "#C0FFEE"
* "color\_text"(site preference kpColorText): "#C0FFEE"
* "color\_text\_secondary"(site preference kpColorTextSecondary): "#C0FFEE"
* "radius\_border"(site preference kpRadiusBorder): "0px"



## Customizing Payment method name

The payment method name “*Klarna Payments*” may be customized via the “[*Merchant Tools*](https://klarna01-tech-prtnr-eu03-dw.demandware.net/on/demandware.store/Sites-Site/default/ViewApplication-SelectSite?MenuGroupID=ChannelMenu&ChannelID=ec0265a8f17125672af2bccdfc&SelectedSiteID=ec0265a8f17125672af2bccdfc&csrf_token=g31fPv4Ab97ROXmU4w09nERj_jgY9hVfcuxcSSByBknXMCIWTuuXq-LbSknRne1wPkIZOLdVo4MARBFkHfTFNVKwX_PrRHv64jUwG92VwmI2gpnBtSFeL7k6QNICz3JUeCNARi2N0vC2aouNrbA9n_pMXL3sZiaLOWsgbE9QwNWH2VknlJU)*>*[*Ordering*](https://klarna01-tech-prtnr-eu03-dw.demandware.net/on/demandware.store/Sites-Site/default/SiteNavigationBar-ShowMenuitemOverview?CurrentMenuItemId=orders&csrf_token=g31fPv4Ab97ROXmU4w09nERj_jgY9hVfcuxcSSByBknXMCIWTuuXq-LbSknRne1wPkIZOLdVo4MARBFkHfTFNVKwX_PrRHv64jUwG92VwmI2gpnBtSFeL7k6QNICz3JUeCNARi2N0vC2aouNrbA9n_pMXL3sZiaLOWsgbE9QwNWH2VknlJU)*>  Payment Methods*” section in Business Manager.



The screenshot below shows the “*KLARNA\_PAYMENTS*” method selected and the administrator choosing a language from the drop-down.

# System Extensions

This section describes all cartridge extensions (custom attributes to system objects, site preferences). Once you have installed the cartridge (“Installation” section), you can check and confirm the extensions have been applied to your system.

## System object extensions

### Order

**Order** system object has been extended with the following custom attributes:

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Attribute ID** | **Description** |
| Klarna Payments Order ID | kpOrderID | Holding the Klarna Payments Order ID, if Klarna Payments is used for the specific order |
| VCN Brand | kpVCNBrand | Holding the Klarna Payments Virtual Card Network card Brand |
| VCN CSC | kpVCNCSC | Holding the Klarna Payments Virtual Card Network card csc (encrypted) |
| VCN Expiration Month | kpVCNExpirationMonth | Holding the Klarna Payments Virtual Card Network card expiration month |
| VCN Expiration Year | kpVCNExpirationYear | Holding the Klarna Payments Virtual Card Network card expiration year |
| VCN PAN | kpVCNPAN | Holding the Klarna Payments Virtual Card Network card pan (encrypted) |
| VCN Holder | kpVCNHolder | Holding the Klarna Payments Virtual Card Network card holder name |
| Is VCN Used | kpIsVCN | True if VCN is used for payment of the order, otherwise false |

Table 1. Order system object attributes

### PaymentTransaction

**PaymentTransaction** system object has been extended with the following custom attributes:

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Attribute ID** | **Description** |
| Fraud Status | kpFraudStatus | Klarna Payments order fraud status |

Table 2. PaymentTransaction system object attributes.

### OrderPaymentInstrument

**OrderPaymentInstrument** system object has been extended with the following custom attributes:

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Attribute ID** | **Description** |
| Klarna Payment Category ID | [klarnaPaymentCategoryID](https://klarna01-tech-prtnr-eu03-dw.demandware.net/on/demandware.store/Sites-Site/default/ViewObjectTypeAttribute-Start?ObjectTypeDefinitionUUID=37a26b715b3660fe91c9c8f1dc&AttributeDefinitionUUID=f63e4cf0293216d763580235b0&csrf_token=j7V05wykqD6PtwAak2ZdYkMrOrll9esrwtt7bmznTSxIOT5rhhI4PDY96Gm5zs1f8Ww1SNBNNRypybFObzn_oTSy9S_pIdcSRCe2nMNgQ-SNAZ25_hFxeKtFV-RB9O7TjWlpJzGLyGoWrzry4D5rGDy6BCUbDi-O0AfNh8UaH7CsnYiVEQ4) | Holds ID of Klarna payment category. |
| Klarna Payment Category Name | klarnaPaymentCategoryName | Holds name of Klarna payment category. |

Table 2. OrderPaymentInstrument system object attributes.

## Custom objects

### KlarnaCountries



**KlarnaCountries** custom object holds the corresponding locale for each Klarna Payments endpoint. Even if you have locales that are not supported by Klarna Payments we recommend to make a corresponding entry in the custom object for that locale (you can use one of the other endpoints). Thus, on the billing page of the unsupported locale you will have the Klarna Payments widget showing an appropriate message like below:

The table below describes attributes of the **KlarnaCountries** custom object:

|  |  |  |
| --- | --- | --- |
| **Attribute Name** | **Attribute ID** | **Description** |
| Country Code | country | Two-letter country code. |
| Klarna Locale | klarnaLocale |  |
| Service Credential ID | credentialID | The ID of service credentials for this locale. |
| Shipping Countries | shippingCountries |  |
| Show Shipping Options | showShippingOptions |  |
| Show Shipping Countries | showShippingCountries |  |

Table 3. KlarnaCountries custom object attributes.

## Site preferences

### Klarna\_Payments group

The site custom preferences have been extended with a new group called “Klarna\_Payments”.

The table below describes the preferences within that group:

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Attribute ID** | **Description** |
| Send product\_url and image\_url | sendProductAndImageURLs | If set to true, product\_url and image\_url fields in Klarna Payments create session call will be included in API call, otherwise product\_url and image\_url fields will not be populated. |
| Merchant Reference 2 Mapping | merchant\_reference2\_mapping | The field from SCC order (basket) object that is mapped to merchant\_reference2 field from klarna API request.  Has to be one of the class attributes of SCC LineItemCtnr. Note that for complex data structures result may not always be as expected. |
| Pre-Assessment is on for: | kpPreAssessment | Pre-assessment flag. Comma separated string values with country codes, which should have the pre-assessment flag ON.  For example the following string “CN, AU” would mean the pre-assessment is ON only for China and Australia.  Note: Even if ON for an EU member country, consumer information will not be sent to Klarna until the authorize step. |
| Border Color Preference | kpColorBorder | CSS hex color to be used in Klarna Payments iFrame |
| Border Selected Color Preference | kpColorBorderSelected | CSS hex color to be used in Klarna Payments iFrame |
| Button Color Preference | kpColorButton | CSS hex color to be used in Klarna Payments iFrame |
| Button Text Color Preference | kpColorButtonText | CSS hex color to be used in Klarna Payments iFrame |
| Checkbox Color Preference | kpColorCheckbox | CSS hex color to be used in Klarna Payments iFrame |
| Checkbox Checkmark Color Preference | kpColorCheckboxCheckmark | CSS hex color to be used in Klarna Payments iFrame |
| Details Color Preference | kpColorDetails | CSS hex color to be used in Klarna Payments iFrame |
| Header Color Preference | kpColorHeader | CSS hex color to be used in Klarna Payments iFrame |
| Link Color Preference | kpColorLink | CSS hex color to be used in Klarna Payments iFrame |
| Text Color Preference | kpColorText | CSS hex color to be used in Klarna Payments iFrame |
| Secondary Text Color Preference | kpColorTextSecondary | CSS hex color to be used in Klarna Payments iFrame |
| Border Radius Preference | kpRadiusBorder | Size (in pixels) of the border radius to be used in Klarna Payments iFrame |
| Attachments | kpAttachments | Flag to swicth on/off the using of attachments when creating a session. Default is OFF. |
| Not available message on billing page | kpNotAvailableMessage | The Klarna Payment not available message on billing page. JSON string holding country code and corresponding message string.  For example:  {      **"GB"**:"Klarna Payment not available",    **"default"**:"Klarna Payment not available" } |
| Virtual Card Network Enabled | kpVCNEnabled | If this option is set to TRUE, SFCC will create a Virtual Card Network settlement for every Klarna order.  Note: the option will only work if VCN private/public keys are configured properly (see vcnPrivateKey and vcnPublicKey below). |
| VCN Private Key | vcnPrivateKey | SSL private key is used by SFCC to decode Virtual Card information (used with kpVCNEnabled). |
| VCN Public Key | vcnPublicKey |  |

Table 4. Site preferences.

## Services

An HTTP service “klarna.http.defaultendpoint” has been added with “klarna.http.service” profile and service credentials for each country (described in **KlarnaCountries** custom object).

# Installation

Installation of this cartridge is a process consisting of the following steps:

1. Verify package contents.
2. Upload the cartridge to the currently active code version on your environment.
3. Import metadata.
4. Configure Klarna integration settings (preferences).
5. Configure Klarna service and credentials.
6. Add the cartridge to the cartridge path.
7. Test the SFRA front-end and make sure Klarna payment options appear correctly.

Note that this section describes the minimum installation process without going in specifics on how to customize the cartridge. By the end of the section, you should have a working Klarna Payments SFRA cartridge.

## Verify Package Contents

**Klarna Payments SFRA** package contains the following items:

* Cartridge called “*Klarna Payments SFRA*” (or simply “*the cartridge*”).
* A site-template archive containing attributes and settings required for the cartridge to work properly.
* This document.

## Cartridge upload

Make sure the “*int\_klarna\_payments\_sfra*” cartridge is uploaded to the active code version of your environment.

## Metadata import

Find the archive ‘*site-template.zip*’ in the installation package, it contains metadata files. Please review the archive, do the necessary modifications, and import it through *BM Administration > Site Development > Site Import & Export* section. The only required modification is renaming the *sites/SFRA* folder of the archive to the ID of your site. If you need to configure more than one SFRA site, just copy-paste the *sites/SFRA* folder.

Check the fields which will be updated after importing file ‘*site-template.zip*’ and be sure that there won’t be any conflict with the existing fields.

Right after metadata import, you should verify all extensions are available within the business manager as described in Section 2. “System Extensions”.

## Service configuration

You need to configure Klarna Payments services credentials.

Go to Credentials tab as shown below:



You need to edit the existing credentials or create new ones using the credentials provided to you by Klarna and the API URL for the specific location (see “Klarna API Information” for the API URLs) as shown below:



## Add cartridge to cartridge path

To activate the cartridge, make sure you prepend “*int\_klarna\_payments\_sfra*” to your cartridge path. This cartridge should be prepended to the “*app\_storefront\_base*” cartridge so it can override base templates and controllers.

## (optional) Enable VCN

In order to enable Virtual Credit Network settlements (VCN settlements, in short), you need to do the following:

1. Generate an SSL keypair (see “Generate SSL Keys for VCN Settlements”).
2. Input the keys from the generated keypair in the site preferences, *vcnPrivateKey* and *vcnPublicKey* respectively.
3. Send the generated public key to Klarna. It will be used to encrypt the VCN pan and csc on our side. Wait for confirmation from Klarna side that the key has been successfully added to your profile.
4. Enable the VCN site preference “*kpVCNEnabled*”.
5. Use a credit-card processor of your choice. By default, this cartridge uses basic\_credit to authorize credit-card settlements.

# Integration with other payment cartridges

The contents of this section is only applicable if there are other payment cartridge integrations within the cartridge path besides **Klarna Payments SFRA**. No matter of the order of those cartridges in the cartridge path, there are certain templates that need to be overwritten by adding a new if-condition and including the right sub-template:

* *\templates\default\checkout\billing\paymentOptions\paymentOptionsContent.isml*
* *\templates\default\checkout\billing\paymentOptions\paymentOptionsSummary.isml*
* *\templates\default\checkout\billing\paymentOptions\paymentOptionsTabs.isml*

Suppose, the website owner requires PayPal as well as Klarna. Each of these templates mentioned above must be copied to a new custom cartridge. The example below shows the new code of the *paymentOptionsContent.isml* template:

<isloop items="${pdict.order.billing.payment.applicablePaymentMethods}" var="paymentOption" status="loopSate">

<isif condition="${paymentOption.ID === 'CREDIT\_CARD'}">

<isinclude template="checkout/billing/paymentOptions/creditCardContent" />

</isif>

<isif condition="${paymentOption.ID === 'KLARNA\_PAYMENTS'}">

<isinclude template="checkout/billing/paymentOptions/klarnaPaymentsContent" />

</isif>

…

**<isif condition="${paymentOption.ID === 'PayPal'}">**

**<isinclude template="paypal/checkout/paypalContent" />**

**</isif>**

</isloop>

Same goes for the other two templates.

# Uninstallation

Uninstallation of the cartridge is:

1. Remove cartridge “*int\_klarna\_payments\_sfra*” from the cartridge path.
2. Remove cartridge code from active code version.
3. Remove cartridge-related metadata from the website.

# Release History

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Changes** |
| 18.1.0 |  | Initial release of Klarna Payments SFRA. |

# Additional Information

## Klarna API Information

The Klarna Payments API is accessible through a few different URLS. There are different URLs for testing and for making live purchases as well as different URLs for depending on if you are based in Europe or in the U.S.

### Live environment

The API for the European production environment can be found at

* <https://api.klarna.com/>

The API for the U.S. production environment can be found at

* <https://api-na.klarna.com/>

### Testing environment

The API for the European testing environment can be found at

* <https://api.playground.klarna.com/>

The API for the U.S. testing environment can be found at

* <https://api-na.playground.klarna.com/>

## Generate SSL Keys for VCN Settlements

In order to generate an RSA keypair with a 4096 bit private key you can use the following *openssl* command:

**openssl genpkey -algorithm RSA -out private\_key.pem -pkeyopt rsa\_keygen\_bits:4096**

In order to extract the public key from an RSA keypair, you can use the following *openssl* command:

***openssl rsa -pubout -in private\_key.pem -out public\_key.pem***

In the folder where you have executed the above commands two new files will be created - public\_key.pem and private\_key.pem.

The contents of the files should look something like:

public\_key.pem

-----BEGIN PUBLIC KEY-----

MIICIjANBgkqhkiG9w0BAQEFAAOCAg8AMIICCgKCAgEAoNYG7l2G8nZa+22oBYZk

tV228lw3UE9WO4oxfknJtKEdHn84x55ULt8KQTh9NVtdeKC8nTfTgyvMt/GNCa18

xuZV/lGYDftKt85hbV5EjOum+StAIufEXvlBX7nMOMc1KyWm9kp2kbqd88mFIX63

KV94OoNEXcNatRDFYR+qz53+ifadDQtQ1slVNStdroCZDJ1+LxtBy9V+BdmsBK1E

RLsKh/JLXyWE24FJKV+z00s7TQkdWW/5ET12OGQYZsWo1yqgi9HplNvrisve8vWP

xaL4m8iZ3I/9yYdg7yANQbTxSJcbbRCgaaagPo30CNxeqU6qafY5g8vY3E52CoXH

DdO4UslX1qcuYIDhqaDzey6W+b8m755xLi+rqQyM4PBWL0J0dM3FVid8+4YKILex

3AKBFciqRCMHSOGaEeyrXKTjlAsghr9RS8PifvQRrL440cHzqw2vX0DvpjSWcmUJ

tW4wUq5RNSsobrxnVmoV6fj1z67Q/1P+l5Ie+oowdahR5ztVqJlO+2PNoX4I5VDs

/Pkz3f8wWVc3Mp2oNT244o+/NIiyRfPFaJJx7JAgrcvZt2nFAmY4QApXLFJCpgEM

wYucE4AH4gJKsh3KZbxRERrrO72bL2rxvWqBp/0h7DcMsV9sQs4BvxxIl6CF506F

ThzmclaKLBAyd5LALiXiPfkCAwEAAQ==

-----END PUBLIC KEY-----

private\_key.pem

-----BEGIN PRIVATE KEY-----

MIIJQQIBADANBgkqhkiG9w0BAQEFAASCCSswggknAgEAAoICAQCg1gbuXYbydlr7

bagFhmS1XbbyXDdQT1Y7ijF+Scm0oR0efzjHnlQu3wpBOH01W114oLydN9ODK8y3

8Y0JrXzG5lX+UZgN+0q3zmFtXkSM66b5K0Ai58Re+UFfucw4xzUrJab2SnaRup3z

yYUhfrcpX3g6g0Rdw1q1EMVhH6rPnf6J9p0NC1DWyVU1K12ugJkMnX4vG0HL1X4F

2awErUREuwqH8ktfJYTbgUkpX7PTSztNCR1Zb/kRPXY4ZBhmxajXKqCL0emU2+uK

y97y9Y/FovibyJncj/3Jh2DvIA1BtPFIlxttEKBppqA+jfQI3F6pTqpp9jmDy9jc

TnYKhccN07hSyVfWpy5ggOGpoPN7Lpb5vybvnnEuL6upDIzg8FYvQnR0zcVWJ3z7

hgogt7HcAoEVyKpEIwdI4ZoR7KtcpOOUCyCGv1FLw+J+9BGsvjjRwfOrDa9fQO+m

NJZyZQm1bjBSrlE1KyhuvGdWahXp+PXPrtD/U/6Xkh76ijB1qFHnO1WomU77Y82h

fgjlUOz8+TPd/zBZVzcynag1Pbjij780iLJF88VoknHskCCty9m3acUCZjhAClcs

UkKmAQzBi5wTgAfiAkqyHcplvFERGus7vZsvavG9aoGn/SHsNwyxX2xCzgG/HEiX

oIXnToVOHOZyVoosEDJ3ksAuJeI9+QIDAQABAoICACRkaUsUNI22RB3yEPu3DiCP

pO6v+QAeA4gTW+GUdqR9dCZLaSCZ7bhxVVOuoX4qPzslO6hjUmOyzG6upFgVPk+P

HNQfyEUZoC148Eib9OziAXUN2URMpv1KbwVm+BO814X8zguai7uru0PHTG1oy677

4Ct1OknxAxxHQDIaxT6XJFo5SA4EinUfNz2Bo3/xry/QjxW/mCK0GwDd4PNp9TGM

FPTv2SgdSDOWzGQlOH5N3owuzMpI8NV6z74wv+i5Ptv41Dzu8WhyXpiYSsk00SRK

HPC68j2bAzTPghp5aSZ9976SGm2SPonJXyboXdiHbI/osdyqDxeIT3iB9GmrHX/i

kHPGJCh7fRZvqj39Hc+IxYjabwW3rDeDIPB7ab9z1KLF4z1D6AZOKCPyTaDRdQ1Q

eDi7LwDmk7NHEPrmF/nIcguQdqbIbmFO2zEs0TOe6y4uBMndRsbQprTNSMUdBkrA

lNaYVSTQ1Z0Y/8DZDpGcyS1OnJv74F15uDjKN6/ov991mZ1JrZ+V2sdS3EDUlmvP

6thQKwI7Ln6h+ApHtWUG1NmvQe5gJE0qAeJ9b45clUzIRUwhVmEp8NoIJh0kAjaN

d4lk7xy9ZRDUY5yekPeYrJPShjsHAyEoktJIjRufI2UUq3uxNjjICoQcOVGfNDIS

YTTPwpu1pmC0C+rh2fgBAoIBAQDRultRArvtc2JKhVOUyZk88zd9kvrI6fNiyKmi

HgiWf7qkTPD9xhOQWDw3iwRFQAD+YkgV5MCBO8wp8oO8GEsOCI+XZWExOcPT0Vfj

PZHiQrTFnlfG/+fAO14xLf3j3ED4YQXdHOKI3xoLknQx/EydLoctxgkkpgWLrsA7

DwdSAg1/0sBvaHY27ogAfdimHdaKZ5OAe4a9k1qP3xVZBuOe8Sd65unBavUJLDuv

ikeNmkSVgW1sm55/729JIr63USHF76It+vE1cdZ+vKg5vYotsQgPzvNBmUO/E8Gj

zMXQRfqfvEDlNXEX0rCupTkw1G6AGTwQc/NPzyr/LTpLe6UBAoIBAQDEUjTiG11V

hf7WjdG3gctRlr+mYapQHgXdVLx2QSaqUYid+0QXK11YfJlsRB6nwa+OED83RfP0

lIFqxpzudSLPmoDuIBT7Dl5c/aleyKs/siUusP8QVDXk6OAR84XSytC35sIRV7pE

VMuBL91jfkQ0Lf/PreslK/kI6Yvwwp4qrHK6/f9TgciHclYtf+/oti4ky6GJgfmP

fmuCqjxmUKbXXFPd5RbL2THGOowilb8zDLjf3RlbjlQFqogAk6H9hp2V0VZLiJHp

UWM3z3zxDWeDaqJ08sHuk/rA9QpsVTu8IGTQsxdj8JwluN1Q+YZiOuPiSENBqPzT

V3exexzo3sD5AoIBAGU3qEyPojz1+9D1SaI8LW2CABzlq4z9g84ABAZOslxX5q7W

x1PinZyDSQSRXg1B13jt29ZdIR79ygnQlg1YOBjcvtgVQHPuafk3RlBQbbCh+vaI

9dn/tUxMGqhnhunKaby1rovJHfdqnPpKwzNAjYUqaGkJ822xhmmke/fEyAanIPa4

stDRvIPEWPTLx5xcOCdx13khpKSnkgRvaLEfpwkVX7Vr7hK/2OSFaYTNmrzXYBQ7

c6D/9d3Oo4nLb/mu+Tq67S19t53Qg/GEgTfkpuRoVPi0KyhUnKKCGWlBMZLTwyIG

S9eTFDKoJ0cSTGipjW7bPua93wZ8eEbRABpf4QECggEANNhQBeEJ0aCdBVHtdrEI

crDaa8X0W1aJi5dol4hYCRajaKsfHAF/QfdgMQVxHwUC5YG4En/Q+DAVWhGWYpXD

RhC3zeFy5FVszyk0sx/fAOlKGvRn5BRW4YRR9GMRzbjsT+RcruBnckdE9ERXGpX9

c/JB3rxZBIt+oIiFM8yfWKtMwsrmNKtFuDftvJeok4KejycFF4eWDqsf828xjPT+

xA/FP4CQD1UqkcpmuFSIwAwXo6LXVY7NTS0nKMiUnTLkLlTIHtLnO9+9jmNapWRP

Tc+hZUuHKlpI8DHFmX2j87LgkFD05eD5lynY4RgZtU1W1C1RdVYwoA72WB7knEaB

uQKCAQAH9s67P/7fFX9dfEans3PHU4nGjD8dJ8eoNQ6DhBMydZpGWI5ZUeEBZDRk

0cBOeRs5BOcS43Em9kETpzawyCwxmnwzl+CzoPzMQcTw9tXomF9HG6RJ9XBdJfGA

ALAwCd4bASxmFM6guSP5GKnZ9aY3tR3tWWDfr7f9z8wOewzzpPclwRh009fPe4TC

NXoEm1MELJVeUieDSLKZgjgCw8WHGqLItONpA0/fwSM2gIcxETVV7qx3aPuJzCVh

LQZoBLQk3UMKsWDdpzeBdiERe66NAgVk92Xe7SY9EY2vymaq761i1x1vlprT27qp

240LDJawqM0IraKmdCvWjofWSaOU

-----END PRIVATE KEY-----