



Saferpay

For Salesforce Commerce Cloud
B2C Commerce

SFRA LINK Integration

Contents

- 1 Summary 3
- 2 Component overview 3
 - 2.1 Functional overview 3
 - 2.2 Use cases 3
 - 2.3 Limitations, Constraints 4
 - 2.4 Comptability 4
 - 2.5 Privacy, Payment 4
- 3 Implementation guide 4
 - 3.1 Setup of Business Manager 4
 - 3.1.1 Setup Storefront cartridges 4
 - 3.1.2 Setup Business Manager cartridges 5
 - 3.1.3 Import Data 5
 - 3.2 Configuration 6
 - 3.2.1 Configure Site Preferences 6
 - 3.2.2 Configure Services 9
 - 3.2.3 Configure Payment Methods 9
 - 3.2.4 Configure Wallets 9
- 4 Testing 10
 - 4.1 Checkout 10
 - 4.1.1 Checkout as a guest 10
 - 4.1.2 Checkout as an authenticated customer with a new card 12
 - 4.1.3 Checkout as an authenticated customer with an existing card 13
 - 4.2 My account 14
 - 4.2.1 Add a credit card to your profile 15
 - 4.2.2 Remove a credit card to your profile 15
 - 4.2.3 Edit a credit card on your profile 16
- 5 Operations, Maintenance 17
 - 5.1 Data storage 17
 - 5.2 Availability 17
 - 5.3 Support 17
- 6 User guide 18
 - 6.1 Roles, Responsibilities 18
 - 6.2 Business Manager 18
 - 6.2.1 Customer Service Center Actions 18
 - 6.2.2 Performing a payment capture 18
 - 6.2.3 Performing a payment cancel 18
 - 6.2.4 Performing a payment refund 19
 - 6.2.5 Jobs 20
- 7 Know issues 20
- 8 Release history 20

1 Summary

This LINK cartridge is the official Saferpay Commerce Cloud LINK integration.

You can find more information about Saferpay here: <https://docs.saferpay.com>. It is required to contact Saferpay and acquire a license before going live. There is always an option to create a free test account here: <https://test.saferpay.com/BO/SignUp>.

More information about the JSON API: <https://docs.saferpay.com/home/interfaces/payment-api>

More information about general integration:

<https://docs.saferpay.com/home/integration-guide/introduction>.

This is handy to get a general idea of what Saferpay is capable of.

The was built and tested with SFRA version 4.4.1 and B2C Commerce version 20.5 (Compatibility Mode: 18.2).

2 Component overview

2.1 Functional overview

Saferpay provides an easy way to integrate with a wide range of payment methods and wallets. The integration makes use of REST API calls to Saferpay. Commerce Cloud will then handle the responses and will process orders.

It is also possible to handle payment captures, cancels and refunds in the Commerce Cloud Customer Service Center.

The following payment methods are supported by default:

- Visa
- Mastercard
- American Express
- Bancontact
- iDEAL
- Diners Club International
- Bonus Card
- PayPal
- Alipay
- TWINT

The following wallets are supported by default:

- Apple Pay
- Masterpass

2.2 Use cases

- As a customer, complete a checkout processes using the Saferpay payment methods with the ability to remember your card data when logged in.
- As a customer, add, edit and remove payments instruments in my account.
- As the merchant, view the Saferpay payment details in the Business Manager.
- As the merchant, perform custom actions regarding the payment (capture payment, cancel payment, refund payment).
- As the merchant, configure multiple settings in the custom site preferences.

2.3 Limitations, Constraints

The plugin was built to work on the SFRA demo site.

The following Saferpay API's are supported:

- Payment Page Interface
- Transaction Interface
- Secure Card Data Interface

The following Saferpay API's are not supported:

- Batch Interface
- OmniChannel Interface
- Saferpay Secure PayGate API

2.4 Comptability

Sitegenesis: Not supported

SFRA version: 4.4.1

Compatibility mode: 18.2

2.5 Privacy, Payment

Credit card data is never processed on the Commerce Cloud side.

New payments will be entered and processed on the Saferpay Payment Page.

There is also a possibility to use the Secure Card Data feature. This feature will allow the customer to remember and reuse the card data entered at a later stage. Card data is always stored on the Saferpay environment, never in Commerce Cloud. Commerce Cloud however stores a reference to the existing card data. You can find more information here:

<https://docs.saferpay.com/home/integration-guide/licences-and-interfaces/secure-card-data>

3 Implementation guide

The LINK integration was developed using Visual Studio Code together with the prophet plugin.

Open the project follow the steps:

- Make sure that node and npm are installed
- Go to the project root and enter "npm install" in the terminal
- After the install is complete, enter "npm run compile" to compile the css and js
- Upload the cartridges to the Business Manager

3.1 Setup of Business Manager

3.1.1 Setup Storefront cartridges

The first thing to do is to setup the Storefront cartridges. The plugin exists out of 3 storefront cartridges. These cartridges will need to be added to Cartridge Path of the required Storefront Site(s).

Name	Purpose
app_storefront_saferpay	Fixes a bug that enforces creditcard validation in the controller. This in not yet released. But it is important to include this cartridge to make the SFRA demo site work.
int_saferpay_sfra	Contains all the business logic pertaining order and payment management through the Saferpay API.
plugin_saferpay_sfra	Contains Payment and Checkout controllers.

It is important to add the cartridges in the following order. Insert these cartridges *before* **app_storefront_base** cartridge.

Cartridge List

plugin_saferpay_sfra:int_saferpay_sfra:app_storefront_saferpay

3.1.2 Setup Business Manager cartridges

When this is done, it is time to setup the Business Manager cartridges. There are 2 cartridges that you will need to be added to the cartridge path for the Business Manager.

Name	Purpose
int_saferpay_sfra	Contains all the business logic pertaining order and payment management through the Saferpay API.
bm_saferpay	<ul style="list-style-type: none">• Contains a job to check for orders that have not been completed after 24 hours (configurable) and fails them. If by any chance the payment flow was completed for this order but failed, the order will be set to the complete status.• Contains custom Customer Service Centre actions per order. These actions will help maintaining the order.
app_storefront_base	Add this cartridge to provide the necessary business logic to the custom actions in the Customer Service Center

It is important to add the cartridges in the following order.

Cartridge List

bm_saferpay:int_saferpay_sfra:app_storefront_base

3.1.3 Import Data

The project also contains a **data** folder. This data folder contains jobs, custom objects, custom attributes, services, payment processors and payment method required.

To import the data, follow the next steps:

- Import jobs
 1. Go to **Administration** → **Operations** → **Import & Export**
 2. Under **Import & Export Files** click on **Upload**
 3. Upload the **jobs/saferpay-jobs.xml**
 4. Go back
 5. Under **Jobs** click on **Import**
 6. Import the uploaded file
- Import custom objects and custom attributes
 1. Go to **Administration** → **Site Development** → **Import & Export**
 2. Under **Import & Export Files** click on **Upload**
 3. Upload the **metadata/saferpay-system-object-type-extensions.xml**
 4. Go back
 5. Under **Meta Data** click on **Import**
 6. Import the uploaded file
- Add the payment processor manually
 1. Go to **Merchant Tools** → **Ordering** → **Payment Processors**
 2. Click on **New**
 3. Fill in the ID: **SAFERPAY_ECOM_DEFAULT**
 4. Click on **Apply**

- Import payment methods
 1. Go to **Merchant Tools** → **Ordering** → **Import & Export**
 2. Under **Import & Export Files** click on **Upload**
 3. Upload the **payment/payment-methods.xml**
 4. Go back
 5. Under **Payment Methods** click on **Import**
 6. Import the uploaded file
- Import the services
 1. Go to **Administration** → **Operations** → **Import & Export**
 2. Under **Import & Export Files** click on **Upload**
 3. Upload the **services/saferpay-services.xml**
 4. Go back
 5. Under **Services** click on **Import**
 6. Upload the imported file

3.2 Configuration

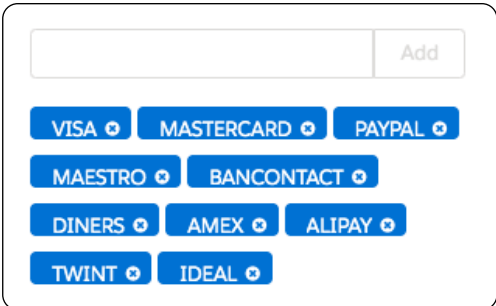
3.2.1 Configure Site Preferences

After importing the data, the custom site preferences become available.

These are all the custom site preferences that all configurable. Make sure to create an API user in the Saferpay backoffice.

1. Go to **Merchant Tools** → **Site Preferences** → **Custom Preferences**.
2. If the custom attributes were imported, you should see an attribute group named **Saferpay**.
3. Click on the group and you should see the settings listed below.

ID	Type	Description
saferpayIsBusinessLicenseEnabled	Boolean	Indicates if the Business or Flex License is enabled for your Saferpay account. Further information can be found in the API documentation. Disabled by default. https://docs.saferpay.com/home/master/licensing
saferpayIsSecureCardDataEnabled	Boolean	Dedicated feature switch for SCD (tokenization) functionalities. Disabled by default.
saferpayIsAutoCaptureEnabled	Boolean	Indicates if the Automatic Capture of a payment needs to happen. The option is enabled by default. If the option is disabled, the order will be placed but the payment status will be set to NOT_PAID.
saferpayApiSpecVersion	String	Saferpay API version. Further information on Saferpay API versions (Changelog) can be found in the API documentation: https://saferpay.github.io/changelog/
saferpayTerminalId	String	The terminal your Saferpay account uses. Only one terminal ID is allowed per site. Further information on Saferpay credentials can be found in the API documentation: https://docs.saferpay.com/home/master/common-saferpay-terms-glossary

safepayLogCategory	String	The CustomerId can be found in the Saferpay Backoffice at Settings JSON API BasicVerification. Further information on Saferpay credentials can be found in the API documentation: https://docs.saferpay.com/home/master/common-saferpay-terms-glossary
safepayCustomerId	String	The CustomerId can be found in the Saferpay Backoffice at Settings JSON API BasicVerification . Further information on Saferpay credentials can be found in the API documentation: https://docs.saferpay.com/home/master/common-saferpay-terms-glossary
safepaySecurityLevel	Number	This configuration is the 3DS security check, the default level is 3.0. Level 0 : LiabilityShift = false, Authenticated = false Level 1 : LiabilityShift = false, Authenticated = true Level 2 : LiabilityShift = true, Authenticated = false Level 3 : LiabilityShift = true, Authenticated = true Note: Payment Methods that bypass these checks are: PayPal, Alipay, TWINT, Bonus Card, iDEAL
safepayAllowedPayment Methods	Set of Strings	These are the allowed payment methods customers can use. An example of all the supported payment methods: 
		Remove the payment methods that are not required. Further information can be found in the API documentation. http://saferpay.github.io/jsonapi/#Payment_v1_PaymentPage_Initialize (PaymentMethods parameter)
safepayAllowedWallets	Set of Strings	These are the allowed wallets customers can use. Further information can be found in the API documentation. http://saferpay.github.io/jsonapi/#Payment_v1_PaymentPage_Initialize (Wallets parameter)
safepayConfigSet	String	Contains the payment page config. Further information can be found in the API documentation. http://saferpay.github.io/jsonapi/#Payment_v1_PaymentPage_Initialize (ConfigSet parameter)
safepayCssUrl	String	Contains the CSS Url. Further information can be found in the API documentation. http://saferpay.github.io/jsonapi/#Payment_v1_PaymentPage_Initialize (CssUrl parameter)

saferpayScdCssUrl	String	Contains the CSS Url for the Alias Insert. Further information can be found in the API documentation. http://saferpay.github.io/jsonapi/#Payment_v1_Alias_Insert (CssUrl parameter)
--------------------------	--------	--

3.2.2 Configure Services

It is necessary to configure the correct Saferpay service credentials.

1. Go to **Administration** → **Operations** → **Services**.
2. If the import was successful, you should see the following services.

Saferpay.AliasAssert	HTTP	Saferpay	Saferpay
Saferpay.AliasDelete	HTTP	Saferpay	Saferpay
Saferpay.AliasInsert	HTTP	Saferpay	Saferpay
Saferpay.AliasUpdate	HTTP	Saferpay	Saferpay
Saferpay.AssertPayment	HTTP	Saferpay	Saferpay
Saferpay.AuthorizeTransaction	HTTP	Saferpay	Saferpay
Saferpay.CancelTransaction	HTTP	Saferpay	Saferpay
Saferpay.CaptureTransaction	HTTP	Saferpay	Saferpay

3. Go to the **Saferpay Credentials**.
4. Fill in the correct URL.
5. Edit the credentials with an API User that was created in the Saferpay backend.

3.2.3 Configure Payment Methods

Validate that the payments methods where imported correctly.

1. Go to **Merchant Tools** → **Ordering** → **Payment Methods**.
2. There should be a list that contains the following payment methods.

ALIPAY	Alipay	Yes
AMEX	American Express	Yes
BANCONTACT	Bancontact	Yes
BONUS	Bonus	Yes
DINERS	Diners Card	Yes
IDEAL	iDeal	Yes
MASTERCARD	Mastercard	Yes
PAYPAL	PayPal	Yes
TWINT	Twint	Yes
VISA	Visa	Yes

3. These payment methods should be linked to the **SAFERPAY_ECOM_DEFAULT** payment processor.

It is possible to change the **Name** and **Image** of the **Payment Methods** but is important that the **ID** is not changed. This **ID** will be used as a reference to the **Saferpay Payment Methods**.

Saferpay Payment Method images are located here:

There is also a custom attribute available to activate pre-authorization. You can find this setting on the payment method details. This attribute is disabled by default. Pre-authorization will overrule the **saferpayIsAutoCaptureEnabled** setting in the business manager. Orders paid with a pre-authorization method will always be AUTHORIZED instead of CAPTURED.

Make sure the payment method supports pre-authorization!

3.2.4 Configure Wallets

Wallets are an optional setting that can be configured in the Custom Site Preferences. The payment page will provide an extra option to pay with the configured wallets if they are provided.

Follow the following steps to add one or more Wallets to the configuration.

1. Go to the **saferpayAllowedWallets** attribute

2. The desired wallets to the Set of Strings (For example **APPLEPAY**)
3. Done

4 Testing

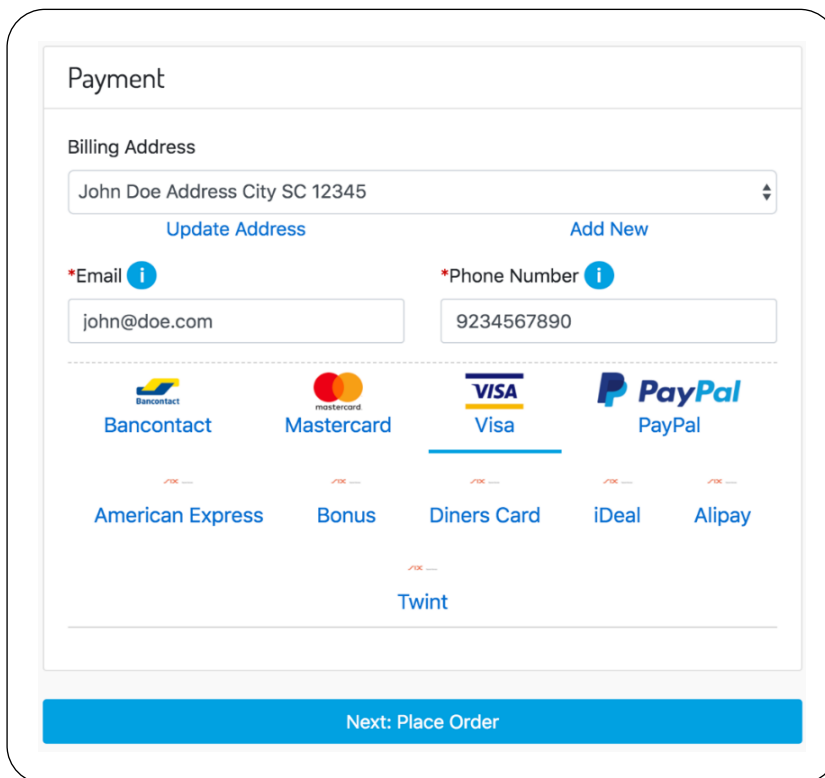
Configure the backend accordingly. Make sure the following Site Preferences are configured according to your Saferpay account:

1. saferpayIsBusinessLicenseEnabled
 - Set to true when you have the Saferpay Business License.
2. saferpayIsAutoCaptureEnabled
 - Set to true when you want to automatically capture orders after the checkout.
3. saferpayIsSecureCardDataEnabled
 - Set to true when you want to enable the Secure Card Data feature.

4.1 Checkout

4.1.1 Checkout as a guest

1. Add a product to your cart.
2. Continue to payment step
3. Select a payment method and click on “Next: Place Order”.



The screenshot shows a mobile payment interface titled "Payment". At the top, there is a "Billing Address" field containing "John Doe Address City SC 12345", with "Update Address" and "Add New" links below it. Below the address are two input fields: "*Email" with the value "john@doe.com" and "*Phone Number" with the value "9234567890". A dashed line separates the contact information from the payment method selection area. This area displays logos for Bancontact, Mastercard, Visa (which is selected with a blue underline), and PayPal. Below these are logos for American Express, Bonus, Diners Card, iDeal, and Alipay. At the bottom of the selection area is the Twint logo. A large blue button at the very bottom of the screen reads "Next: Place Order".

- Validate the order and click on "Place Order".

Payment
Edit

Billing Address:
 John Doe
 Address
 City SC 12345
 john@doe.com
 9234567890

Payment:
 VISA

Place Order

- The checkout will now redirect to Saferpay.

<p>Total amount</p> <p style="font-size: 1.2em; font-weight: bold;">USD 197.38</p> <p>Offer</p> <p>Description of payment</p> <p>Merchant</p> <p>Forward Kempische Steenweg 311 3500 Hasselt Belgium ruben.bielen@forward.eu +32479198515</p>	<p>Card data</p> <p>Card number * 9010 0031 5000 0001</p> <p>Valid until * 05/2025</p> <p>Owner name * John Doe</p> <p>Card verification code * 392</p>	
Back	Cancel	PAY

- Fill the details and continue the payment.
- Now an order is created in the Business Manager.
 If "saferpayIsAutoCaptureEnabled" is enabled, the order will be captured automatically.

General
Attributes
Payment
Notes
History

Payment Information for Order '00029904'

Order Total:	\$2,221.78
Amount Paid:	\$0.00
Balance Due:	\$2,221.78
Invoice Number:	00060504
Payment Status:	Paid
Payment Method:	VISA

Processor: SAFERPAY_ECOM_DEFAULT
 Transaction: 00029904
 Amount: \$2,221.78

saferpayCaptureId: d8nAjbO3SxEtA1zhWO6A9v1x81b_c
 Temporary token: m70q7jbydqks3x5an175ywktp
 saferpayTransactionId: K23GbrA33KbpUAxnUGptb6Mr7xYA
 saferpayConfirmationType: TRANSACTION
 saferpayTransactionStatus: CAPTURED
 saferpayLiabilityShift: true
 saferpayLiabilityAuthenticated: true

4.1.2 Checkout as an authenticated customer with a new card

1. Add a product to your cart.
2. Continue to payment step.
3. Select a payment method. If the `saferpaylsBusinessLicenseEnabled` and `saferpaylsSecureCardDataEnabled` settings are enabled, the option to save a card for later use will become visible.

The screenshot shows a payment form titled "Payment". It includes a "Billing Address" dropdown menu with the text "John Doe Address City AL 12345", and buttons for "Update Address" and "Add New". Below this are input fields for "*Email" (john@doe.com) and "*Phone Number" (9234567890). A grid of payment methods is displayed, including Bancontact, Mastercard, Visa (selected), PayPal, American Express, Bonus, Diners Card, iDeal, Alipay, and Twint. At the bottom, there are radio buttons for "Use new card" (selected) and "Save card for later use". A blue button at the very bottom says "Next: Place Order".

4. Enable the "Save card for later use" option.
5. Click on "Next: Place Order".
6. Validate the order and click on "Place Order".

The screenshot shows a payment summary titled "Payment" with an "Edit" link. It lists the "Billing Address" as "John Doe", "Address", "City AL 12345", "john@doe.com", and "9234567890". Below this, it shows "Payment:" followed by "VISA".

7. The checkout will now redirect to Saferepay

The screenshot shows a payment form with the following sections:

- Total amount:** USD 197.38
- Card data:**
 - Card number *: 9010 0031 5000 0001
 - Valid until *: 05/2025
 - Owner name *: John Doe
 - Card verification code *: 392
- Merchant:** Forward, Kempische Steenweg 311, 3500 Hasselt, Belgium, ruben.bielen@forward.eu, +32479198515
- Buttons:** Back, Cancel, and a large blue PAY button.

4.1.3 Checkout as an authenticated customer with an existing card

1. Add a product to your cart.
2. Continue to payment step.
3. Select a payment method. If the `saferepaysBusinessLicenseEnabled` and `saferepaysSecureCardDataEnabled` settings are enabled, there is an option to select an existing card.

The screenshot shows a payment form with the following sections:

- Payment:** Billing Address dropdown (John Doe Address City SC 12345), Update Address, Add New.
- Contact Information:** *Email (john@doe.com), *Phone Number (9234567890).
- Payment Methods:** Bancontact, Mastercard, Visa (selected), PayPal, American Express, Bonus, Diners Card, iDeal, Alipay, Twint.
- Card Selection:** VISA, Card Number: xxxx xxxx xxxx 0001, Expiration: 05/25, Use new card, Save card for later use.
- Next Step:** Next: Place Order button.

4. Click on “Next: Place Order”.

5. Validate the order and click on "Place Order".

Payment [Edit](#)

Billing Address:
John Doe
Address
City AL 12345
john@doe.com
9234567890

Payment:
VISA

[Place Order](#)

6. The checkout will now execute the payment directly without entering any details.

4.2 My account

This will only be available if the `saferpayIsSecureCardDataEnabled` is set to true!
The customer has the possibility to view, add, delete or edit their credit card data.

There is an overview of the saved cards in the "my account".

Payment [View](#)

Credit VISA
xxxx xxxx xxxx 0001
Ending 05/25

[Add New](#)

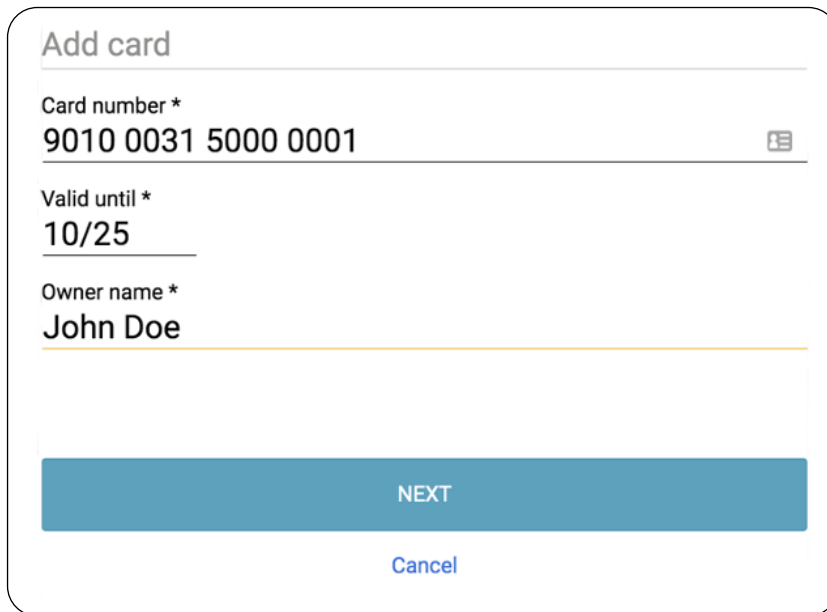
Click on view for a detailed overview.

Credit VISA xxxx xxxx xxxx 0001 Ending 05/25 Edit Payment	×
Credit VISA xxxx xxxx xxxx 0001 Ending 10/25 Edit Payment	×

[Back to My Account](#) [Add New](#)

4.2.1 Add a credit card to your profile

1. Click **Add New**.
2. The page will contain an iFrame with the Saferpay form as described here:
<https://saferpay.github.io/sndbx/CssiFrame.html>



The screenshot shows a form titled "Add card" with the following fields and values:

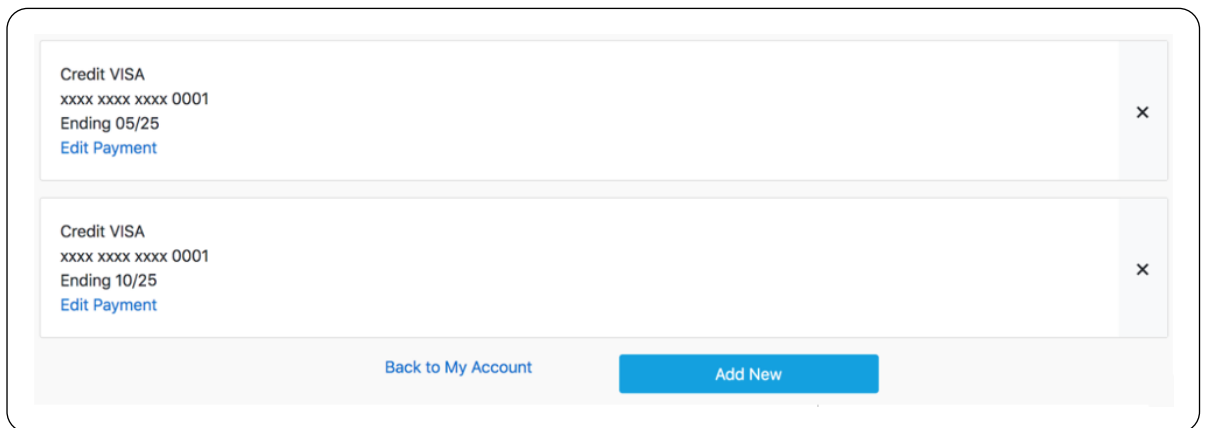
- Card number *: 9010 0031 5000 0001
- Valid until *: 10/25
- Owner name *: John Doe

At the bottom of the form, there are two buttons: a blue "NEXT" button and a blue "Cancel" link.

3. Fill in the card details. Saferpay testcards can be found here:
<https://docs.saferpay.com/home/integration-guide/testing-and-go-live>
4. Click on **Save**.

4.2.2 Remove a credit card to your profile

1. Go to the detailed overview.



The screenshot shows a list of credit cards with the following details:

- Card 1: Credit VISA, xxxx xxxx xxxx 0001, Ending 05/25, Edit Payment, and a close button (x).
- Card 2: Credit VISA, xxxx xxxx xxxx 0001, Ending 10/25, Edit Payment, and a close button (x).

At the bottom of the list, there are two buttons: a blue "Back to My Account" link and a blue "Add New" button.

2. To remove card data, click on the x.

Administration / Operations / Jobs / **Safepay - Check created orders**

General Schedule and History Resources Job Steps Failure Handling Notification

Job Parameters

Scope: **FailArch**

FailExpiredOrders

Select and Configure Step

custom.CSComponents.FailExpiredOrders

Context: Organization, Site

ID*
FailExpiredOrders

Description

ExpireAfterHours*
24 Job Parameters

IsDisabled Job Parameters

Always execute on restart.

Exit Status Rules
Add Rule

4.2.3 Edit a credit card on your profile

1. Click on **Edit Payment**.

Edit Payment

Credit

Card Number ⓘ
xxxx xxxx xxxx 0001

* Expiration Month
05

* Expiration Year
2025

Cancel Save

2. Update the expiration date. It is only possible to edit the date, to edit card data. It is required to add a new credit card.

5 Operations, Maintenance

5.1 Data storage

Data is stored on multiple locations.

- System Objects
 - Profile
 - Secure Card Data information will be saved on the profile in a JSON format. This JSON contains the following information:
 - Brand
 - Payment Method
 - Card Expiration Date
 - Masked Card number
 - The alias Id
 - The date the alias expires
 - The site that the alias was registered
 - PaymentTransaction
 - The PaymentTransaction will contain data so that the payment can be referenced later on.
 - Saferpay Capture Id
 - Saferpay Transaction Id
 - Saferpay Payment Token
 - Saferpay Confirmation Type (PaymentPage or Transaction interface used)
 - Saferpay Transaction Status (CAPTURED or AUTHORIZED)
 - Saferpay Liability Shift status
 - Saferpay Liability Shift authentication status
 - Custom Objects
 - Refund
 - Whenever a refund is performed, a record will be added to the Refund Custom Object. The refund will contain information about the refund.
 - Order ID of the refund
 - Amount
 - Currency used
 - Refund date
 - Refund status
 - Transaction Id
 - Order
 - Order History
 - Communication to and from Saferpay will be logged in the Order History. This way the merchant knows exactly what happened with the payment.

5.2 Availability

When Saferpay services are down the customer will not be able to complete the checkout. Service availability and performance can be checked in the Salesforce backend by the merchant.

5.3 Support

Read more here: <https://test.saferpay.com/>

6 User guide

6.1 Roles, Responsibilities





Integration will require a Commerce Cloud developer that is familiar with SFRA and Commerce Cloud code.

6.2 Business Manager

6.2.1 Customer Service Center Actions

Make sure to assign the correct user rights to the correct roles.

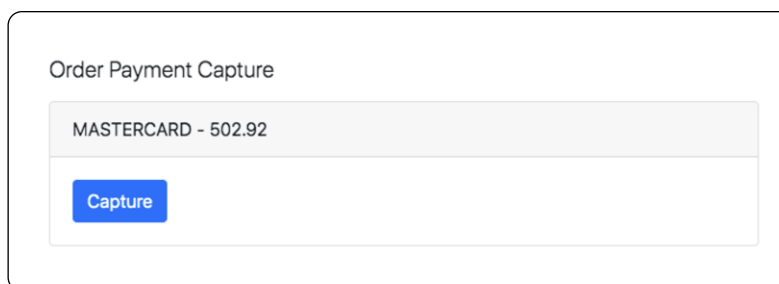
1. Go to **Administration** → **Organization** → **Roles & Permissions**.
2. Click on the on a role.
3. Go to **Customer Service Center Permissions**.
4. Select the specific sites.
5. Enable following roles.

Customer Service Center Module	Module Description
 Order	
 Cancel payment	Cancel payment
 Refund payment	Refund payment
 Capture payment	Capture payment

6.2.2 Performing a payment capture

Order captures are managed by custom actions on the order itself.

1. Press the **Find Order** button.
2. Select an order.
3. Press the **More** button on the top right corner and select **Capture payment**.
Only orders that are OPEN/NEW/COMPLETED and have an AUTHORIZED payment can be captured!



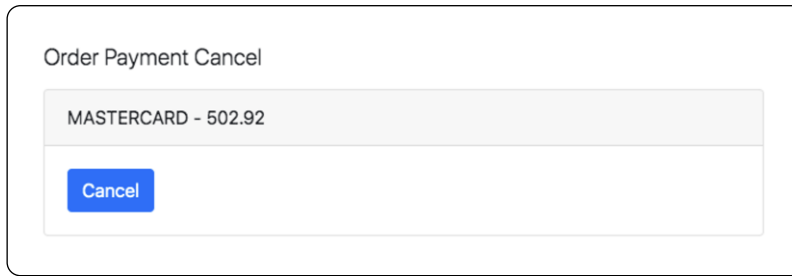
4. Click on **Capture** to refund the order.

6.2.3 Performing a payment cancel

Order cancels are managed by custom actions on the order itself.

1. Press the **Find Order** button.
2. Select an order.

3. Press the **More** button on the top right corner and select **Cancel payment**.
Only orders that are OPEN/NEW/COMPLETED and have an AUTHORIZED payment can be cancelled!



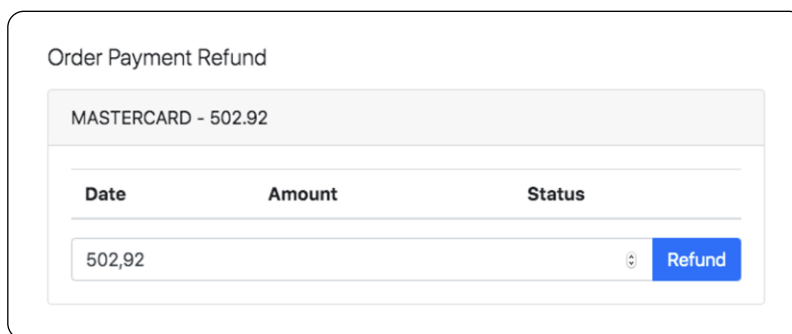
The screenshot shows a dialog box titled "Order Payment Cancel". At the top, it displays "MASTERCARD - 502.92". Below this, there is a blue button labeled "Cancel".

4. Click on **Cancel** to cancel the order.

6.2.4 Performing a payment refund

Order refunds are managed by custom actions on the order itself.

1. Press the **Find Order** button.
2. Select an order.
3. Press the **More** button on the top right corner and select **Refund payment**.
4. The total amount is automatically filled in.
Only orders that are OPEN/NEW/COMPLETED and have a CAPTURED payment can be refunded.

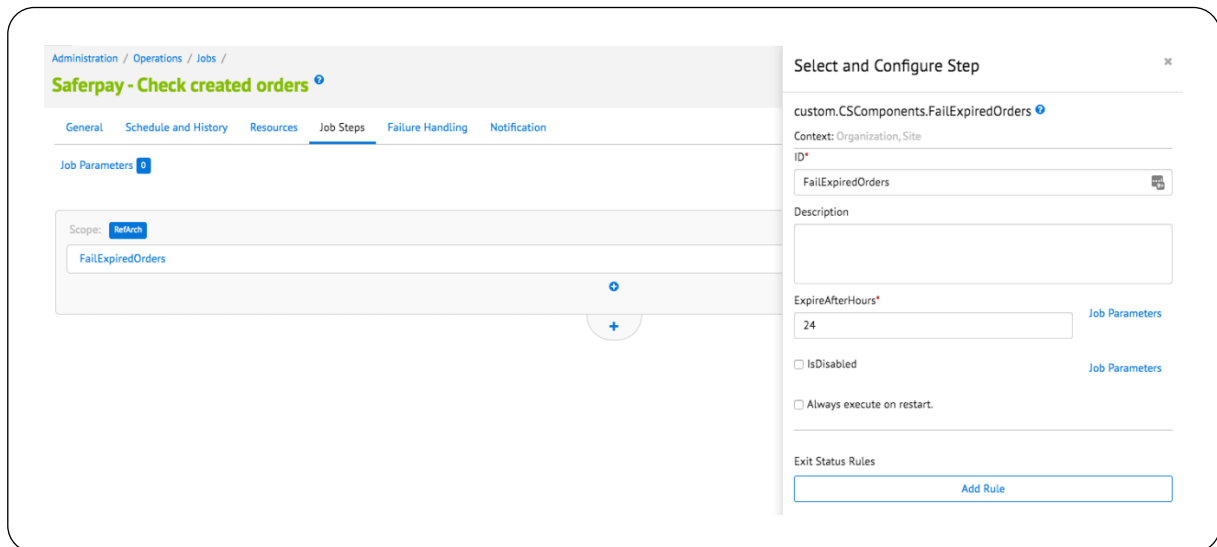


The screenshot shows a dialog box titled "Order Payment Refund". At the top, it displays "MASTERCARD - 502.92". Below this is a table with three columns: "Date", "Amount", and "Status". The "Amount" column contains the value "502,92". To the right of the table is a blue button labeled "Refund".

5. Click on **Refund** to refund the order.

6.2.5 Jobs

There is 1 job available that will handle expired orders. Make sure to change the scope to the required sites.



There will be a Job available to check CREATED orders before a given time. The job will do the following.

1. If the order does not exist is Safepay Fail the order.
2. If the order does exist and is authorised Cancel the transaction in Safepay and Fail the order in Salesforce, the order is expired. If the order does exist and is captured Place the order and set the status of the order to payed.
3. If none of the previous statements is valid for the order Cancel the transaction in Safepay and Fail the order in Salesforce.

There are 2 parameters available:

- ExpireAfterHours
 - The job will collect orders with the CREATED status that are older than the specified number of hours.
- IsDisabled
 - Enable to skip the step.

7 Know issues

No known issues.

8 Release history

Version	Date	Changes
20.1.0	2020-05-06	Initial release
20.2.2	2020-08-18	ClientInfo parameter added to RequestHeader