Cybersource
Storefront Reference Architecture

cybersource
A Visa Solution

Cybersource Contact Information

For general information about our company, products, and services, go to http://www.cybersource.com.
For sales questions about any Cybersource Service, email sales@cybersource.com or call 650-432-7350 or 888-330-2300 (toll free in the United States).
For support information about any Cybersource Service, visit the Support Center: http://www.cybersource.com/support

Copyright

By accepting this document, you acknowledge and accept that you are responsible for and assume liability for the functionality, maintenance and availability of your software and network. At all times, it is your responsibility to ensure the accuracy, technical sufficiency and functionality of your software, network, plug-ins, configurations, applications, code, application program interfaces (APIs), software development kits and all other technology ("Your Network"). You are responsible for Your Network's ability to use and/or access the Cybersource network, any Cybersource API and receive the benefit of Cybersource's services. You are responsible for all costs, fees, expenses and liabilities associated with Your Network's ability to access and interface with the Cybersource network and receive the benefit of Cybersource's services. Cybersource will not be responsible or liable for loss or costs associated with or that results from Your Network's inability to connect to or process transactions on the Cybersource network.

Restricted Rights Legends

For Government or defense agencies: Use, duplication, or disclosure by the Government or defense agencies is subject to restrictions as set forth the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 and in similar clauses in the FAR and NASA FAR Supplement.

For civilian agencies: Use, reproduction, or disclosure is subject to restrictions set forth in subparagraphs (a) through (d) of the Commercial Computer Software Restricted Rights clause at 52.227-19 and the limitations set forth in Cybersource Corporation's standard commercial agreement for this software. Unpublished rights reserved under the copyright laws of the United States.

Trademarks

Authorize.Net, eCheck.Net, and The Power of Payment are registered trademarks of Cybersource Corporation. Cybersource, Cybersource Payment Manager, Cybersource Risk Manager, Cybersource Decision Manager, and Cybersource Connect are trademarks and/or service marks of Cybersource Corporation. Visa, Visa International, Cybersource, the Visa logo, and the Cybersource logo are the registered trademarks of Visa International in the United States and other countries. Bank America and the Bank America logo are the registered trademarks of Bank of America in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

Confidentiality Notice

This document is furnished to you solely in your capacity as a client of Cybersource and as a participant in the Visa payments system.

By accepting this document, you acknowledge that the information contained herein (the "Information") is confidential and subject tothe confidentiality restrictions contained in Visa's operating regulations and/or other confidentiality agreements, which limit your use of the Information. You agree to keep the Information confidential and not to use the Information for any purpose other than its intended purpose and in your capacity as a customer of Cybersource or as a participant in the Visa payments system. The Information may only be disseminated within your organization on a need-to-know basis to enable your participation in the Visa payments system. Please be advised that the Information may constitute material non-public information under U.S. federal securities laws and that purchasing or selling securities of Visa Inc. while being aware of material non-public information would constitute a violation of applicable U.S. federal securities laws.

Release: February 2025
Version: 24.1.3

Contents

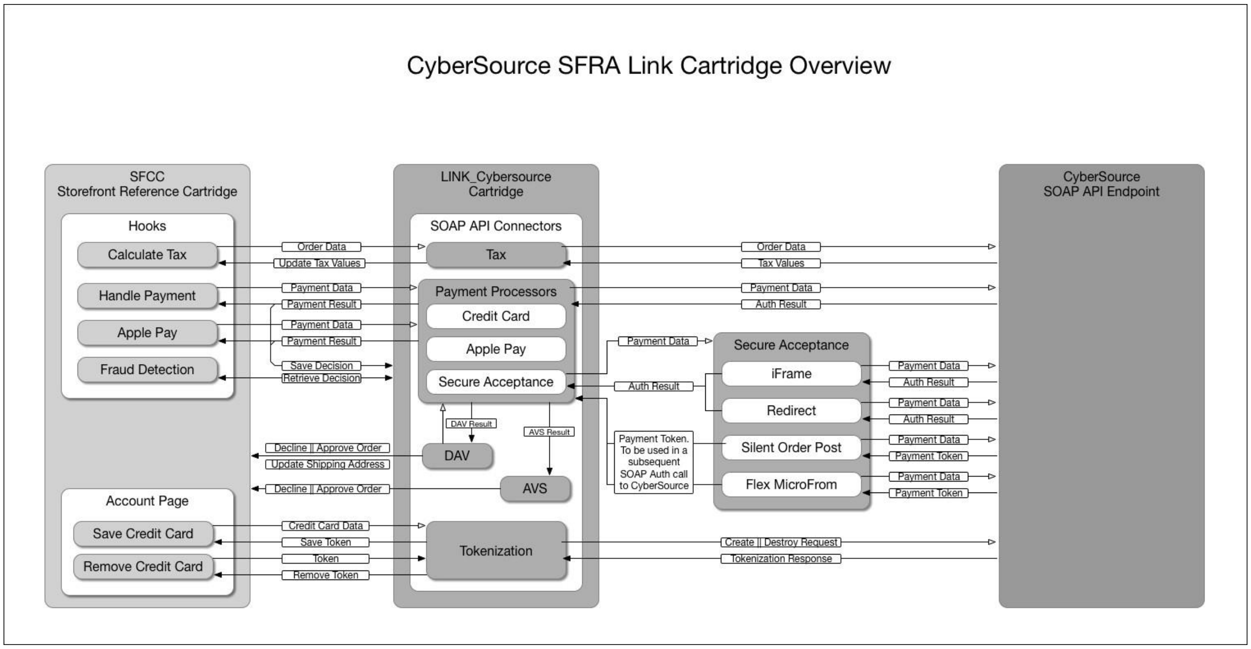
1. Introduction

- **Description:** Cybersource, a Visa solution, is the only global, modular payment management platform built on secure Visa infrastructure with the payment reach and fraud insights of a massive $500B+ global processing network. You can find out more about what Cybersource does here.
- **Categories:** Payment Processing, Fraud Detection, Address Validation, Tax Computation
- **Version:** 24.1.3
- **Compatibility:** This version of the Cybersource cartridge is not compatible with versions of SFRA higher than Release 7.0. This version can be found on the Master branch of the SFRA repository in February 2025. This version is compatible with Salesforce B2C Commerce 21.2 release.

Contact

- Global Partner Solutions - CS: GlobalPartnerSolutionsCS@visa.com

2. Cybersource SFRA Cartridge Architecture



3. Install the Cartridge and Setup Workspace

3.1. Install the Cartridge

Cybersource's Storefront Reference Architecture Cartridge can be installed from SalesforceCommerce Cloud's marketplace link.

3.2. Setup workspace

1. Create a folder “Cybersource” in your workspace and place the cartridge(**int_cybersource_sfra**) downloaded from Marketplace.
2. If you have a different project set-up, you will need to open the file **‘/package.json’** and modify the paths.base value to point to your **‘app_storefront_base’** cartridge. This path is used by the JS and SCSS build scripts.
3. If using VSCode, install the extension Prophet Debugger link or any other SFCC extensionand include below in dw.json ().

```
{
    "hostname": "your-sandbox-hostname.demandware.net",
    "username": "yourlogin",
    "password": "yourpwd",
    "version": "version_to_upload_to",
    "cartridge": [
                "int_cybersource_sfra",
                "app_storefront_base",
                "modules"
 ]
}
```

4. Create a p12 file that you can use to authenticate requests. Place the file generated in the webreferences2 folder in the same cartridge as the WSDL file and the file extension must be jks or pkcs12. Duplicate the CyberSourceTransaction.wsdl file, CyberSourceTransaction.wsdl.properties file and rename them with the same name as your respective p12 file.
   Repeat the above step to use multiple p12 files as per requirements.

Note: If you are currently using cartridge v21.1.0 or earlier or not upgradind to our cartridge version 24.1.3 can refer to **Merchant documentation guide for SFCC Ent P12 authentication.pdf** under cartridge documentation folder to be compliant with p12 update.

NOTE: If you are using different IDE, refer respective developer guide to setup the workspace.

3.3. Build and Upload the code

Prerequisite: Install node under "Cybersource" folder.

Install sgmf-scripts and copy-webpack-plugin

```
npm install sgmf-scripts && npm install copy-webpack-plugin
```

Compile the Code

```
Compile JS and SCSS with following command
npm run compile:js && npm run compile:scss
```

Upload the code to Salesforce Commerce Cloud instance

```
npm run uploadCartridge
```

4. Configure the Cartridge

Prerequisite

If you are new to Cybersource, and would like to start using Cybersource Cartridge quickly, begin by signing up for a Sandbox Account.

You will also need to create a p12 file that you can use to authenticate requests to our sandbox server. Follow same steps to generate P12 file for Production server.

4.1. Setup Cartridge Path

To set up the cartridge path:

1. In the Business Manager, go to **Administration > Sites > Manage Sites > [yourSite] >Settings**
2. For the Cartridges, enter **int_cybersource_sfra:app_storefront_base** and select **Apply**

4.2. Upload metadata

Cybersource cartridge installed from Marketplace comes with metadata to import.

1. Go to **Cybersource/metadata/sfra_meta/sites/** folder.
2. Rename **yourSiteID** folder name with your site ID in Business Manager (this can be found bylooking up **Administration->Sites->Manage Sites**)
3. Zip **sfra_meta** folder.
4. Go to Administration->Site Development->Site Import & Export and upload sfra_meta.zip file.
5. Import the uploaded zip file.

On successful import, it creates following metadata:

- Site Preferences (Cybersource, Cybersource_SecureAcceptance, Cybersource_Paypal, Cybersource_GooglePay, Cybersource_Klarna, Cybersource_VisaCheckout, Cybersource_BankTransfer, Cybersource_PayerAuthentication, Cybersource_WeChat, Cybersource_Tokenization, Cybersource_TaxConfiguration, Cybersource_DeliveryAddressVerification, Cybersource_DeviceFingerprint, Cybersource_Alipay)
- Service (cybersource.conversiondetailreport, cybersource.soap.transactionprocessor.generic, cybersource.http.flextoken, cybersource.assymentrickeymanagement)
- Payment Processor (KLARNA_CREDIT, CYBERSOURCE_ALIPAY, BANK_TRANSFER, CYBERSOURCE_WECHAT)
- Payment Method
- Job (CyberSource: Decision Manager Order Update)

4.3. Cybersource Core (Required)

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/meta/ Cybersource.xml"** in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource** and set values for the following parameters**:**

Field	***Description***
Enable Cybersource Cartridge	Enable or disable Cybersource Cartridge. If disabled none of the Cybersource services are invoked

Cybersource Merchant ID	Cybersource Merchant ID
CsP12_Name	Name of the p12 file added in webreferences2 folder in Step 3.2
CsP12_UserName	Friendly name extracted from p12 (run openssl pkcs12 -in CyberSourceTransaction.pkcs12 -info in terminal to extract friendly name of p12 key uploaded in webreferences2 folder. Use the serialNumber and CN of the friendly name returned. Ex: serialNumber=16903992964110187241 02,CN=sfcc_cybs)
CsP12_Password	Enter the password added while generating p12 key.
Cybersource Endpoint	Select Test (Test) or Production (Production)
Cybersource Developer ID	Unique identifier generated by Cybersource for System Integrator
CsTransactionType	Select Auth/Sale transaction for Credit card
Cybersource Disable Debug logging	Some trace information will be stored in the impex folder on the server with the order ID as the name.
Cybersource Master Card Auth Indicator	Preauthorization: 0 passed in request. Final authorization: 1 passed in request. Undefined authorization: omit authIndicator field from the request message

Table 1: Cybersource Core preferences

4.4 Services (Required)

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/services.xml"** in Business Manager (**Administration > Operations > Import & Export**).

Step 2: Go to **Administration > Operations > Services**

Step 3: Make sure service with name **cybersource.soap.transactionprocessor.generic** exist.

4.5 Payment Processor (Required)

Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/sites/yourSiteID/payment-processors.xml"** in Business Manager (**Merchant Tools > Ordering > Import & Export**).

5. Configure the Payment method

5.1. Credit Card Authorization

Overview

The CC Auth service is integrated via the SFRA OOTB dynamically generated app.payment.processor.cybersource_credit hook. The cybersource_credit hook is registered in the hooks.json file with script ./cartridge/scripts/hooks/payment/processor/cybersource_credit. This script acts as a wrapper to the core Cybersource Authorization code. Behind this wrapper, an API request is constructed, sent to Cybersource, and the response parsed. In the case of a successful authorization (response code 100), the hook returns a JSON object without an error. All other response codes received result in an error being present in the return object, triggering the storefront to display an error message, and not create the order. Actions taken when making the Authorization call are as follows:

1. Creates Cybersource authorization request using ship-to, bill-to, credit card data, and purchase total data from the current basket.
2. If authorize Payer is configured, then make the authorize payer request. If not, ignore and continue with the authorization request.
3. Create a credit card authorization request.
4. If DAV is enabled, set up DAV business rules, as needed.
5. Set up AVS if enabled.
6. Make the service call to Cybersource via the SOAP API.
7. If Delivery Address Verification is enabled, then:
    a. Capture pertinent DAV result information & DAV Reason Code. Update shipping address if a suggestion was returned and the 'CS DAV Update Shipping Address with DAV Suggestion' site preference is enabled.
    b. If DAV fails and DAV On Failure is set to 'REJECT', then exit immediately withrejection response
8. If DAV On Failure is set to 'APPROVE' and the DAV Reason Code is a failure code (not 100), then: a. Exit immediately with declined or review response, as merchant defines
9. Capture pertinent AVS information.
10. Capture Fraud response in a session variable to be handled later.
11. Validate authorization reason code and set corresponding values, based on Authresponse code.

Implementation

Cybersource Cartridge supports the following ways of processing Credit Card

a. Secure Acceptance Hosted Checkout – iFrame
b. Secure Acceptance Redirect
c. Secure Acceptance Checkout API
d. Secure Acceptance Microform
e. Direct Cybersource Payment API

Prerequsite

In the Business Manager, go to **Merchant Tools > Ordering > Payment Methods** andselect **CREDIT_CARD**. And in **CREDIT_CARD details**, double check if **Payment Processor** = **"CYBERSOURCE_CREDIT"**

5.1.1 To Setup Secure Acceptance Hosted Checkout – iFrame

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_SecureAcceptance.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences** > **Custom Preferences>Cybersource_SecureAcceptance** and set values for the parameter:

Field	***Description***	***Value to Set***
CsSAType	Secure Acceptance Type	SA_IFRAME
SA_Iframe_ProfileID	Secure Acceptance Iframe Profile ID. Follow "** Creating a Hosted Checkout Profile **"step from SA Guide url	
SA_Iframe_SecretKey	Secure Acceptance Iframe secret key Follow this link.	
SA_Iframe_AccessKey	Secure Acceptance Iframe Access Key	
CsSAIframetFormAction	Cybersource secure acceptance Iframe form action	
CsSAOverrideBillingAddress	Cybersource Secure Acceptance Override Billing Address	
CsSAOverrideShippingAddress	Cybersource Secure Acceptance Override Shipping Address	
CsCvnDeclineFlags	Cybersource Ignore CVN Result (CVN)	

Table 2: SA Iframe Preferences

5.1.2. To Setup Secure Acceptance Redirect

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/meta/Cybersource_SecureAcceptance.xml"** in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools** > **Site Preferences > Custom Preferences >Cybersource_SecureAcceptance** and set values for the parameter:

Field	***Description***	***Value to Set***
CsSAType	Secure Acceptance Type	SA_REDIRECT

SA_Redirect_ProfileID	Secure Acceptance Redirect Profile ID Follow "** Creating a Hosted Checkout Profile**"step from SA Guide url.	
SA_Redirect_SecretKey	Secure Acceptance Redirect Secret Key. Follow this link.	
SA_Redirect_AccessKey	Secure Acceptance Redirect Access Key. Get the access key from above step.	
CsSARedirectFormAction	Cybersource secure acceptance redirect form action.	
CsSAOverrideBillingAddress	Cybersource Secure Acceptance Override Billing Address.	
CsSAOverrideShippingAddress	Cybersource Secure Acceptance Override Shipping Address.	
CsCvnDeclineFlags	Cybersource Ignore CVN Result (CVN).	

Table 3: SA Redirect Preferences

5.1.3. To Setup Secure Acceptance Checkout API

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/meta/Cybersource_SecureAcceptance.xml"** in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_SecureAcceptance** and set values for the parameter:

Field	***Description***	***Value to Set***
CsSAType	Secure Acceptance Type	SA_SILENTPOST
SA_Silent_ProfileID	Secure Acceptance Silent Post Profile ID. Follow **"Creating Checkout AP Profile"** section from this link.	
SA_Silent_SecretKey	Secure Acceptance Silent Post Secret Key. Follow this link.	
SA_Silent_Access Key	Secure Acceptance Silent Post Access Key. Get the access key from above step.	
Secure_Acceptance_Token_Create_Endpoint	Secure_Acceptance_Token_Create_Endpoint.	
Secure_Acceptance_Token_Update_Endpoint	Secure_Acceptance_Token_Update_Endpoint.	

CsSAOverrideBillingAddress	Cybersource Secure Acceptance Override Billing Address.	
CsSAOverrideShippingAddress	Cybersource Secure Acceptance Override Shipping Address.	
CsCvnDeclineFlags	Cybersource Ignore CVN Result (CVN).	

Table 4: SA SilentPost Preferences

5.1.4. To Setup Secure Acceptance Microform

Prerequisites: You will also need to create an API Key and API Shared Secret Key that you can use to authenticate requests to our sandbox. Follow same steps to generate Production key andshared secret.

Step 1: In Business Manager, go to Administration > Operations > Services and click on the 'CybersourceFlexToken' Credentials. Ensure the appropriate URL is set for the environment you are configuring:
Test: https://apitest.cybersource.com/microform/v2/sessions?format=JWT
Production: https://api.cybersource.com/microform/v2/sessions?format=JWT

Step 2: Navigate to Administration > Operations > Services and click on the 'AssymentricKeyManagement' Credentials. Ensure the appropriate URL is set for the environment you are configuring:
Test: https://apitest.cybersource.com/flex/v2/public-keys
Production: https://api.cybersource.com/flex/v2/public-keys

Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/meta/Cybersource_SecureAcceptance.xml"** in Business Manager (**Administration > Site Development > Import & Export**)

Step 3: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_SecureAcceptance** and set values for the parameter:

Field	***Description***	***Value to Set***
CsSAType	Secure Acceptance Type	SA_FLEX
SA_Flex_HostName	Test: apitest.cybersource.com Production: api.cybersource.com	
SA_Flex_KeyID	Flex Microform Key ID. Follow link to generate keys.	
SA_Flex_SharedSecret	Flex Microform Shared Secret	
SA_Flex_AllowedCardNetworks	Configure card types for Cybersource Microform	VISA MASTER DISCOVER DINERSCLUB JCB

		MAESTRO AMEX CUP JCREW CARTESBANCAIRES ELO

Table 5: SA Flex Microform Preferences

Step 4: Navigate to '**Administration > Global Preferences > Locales**' and ensure the local'en_US' is present. If not present, create a new local with the following information:

- Language Code: en
- Country Code: US
- XML Code: en-US
- ISO-3 Language: eng
- ISO-3 Country: USA
- Fallback Locale: English(en)

Once present, select the 'en_US' local, and navigate to the 'Regional Settings' tab. In the 'LongDate Pattern' field, enter the sting: EEE, dd MMM yyyy HH:mm:ss

Note: Please refer to steps added in **Cybersource for Salesforce B2C Commerce SOAP - Microform v2 Upgrade Steps.pdf** in our documentation section to update the version of Microform without updating to our cartridge version 24.1.3 which adds Microform v2..

5.1.5. To Setup Direct Cybersource SOAP API

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/meta/Cybersource_SecureAcceptance.xml"** inBusiness Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_SecureAcceptance** and set values for the parameter:

Field	***Description***	***Value to Set***
CsSAType	Secure Acceptance Type	None

Table 6: SA None Preferences

5.1.6. Payer Authentication (3D Secure)

Prerequisite

- Please contact your Cybersource Representative to sign up and receive your PayerAuthentication credentials.
- Go to **Merchant Tools > Ordering > Payment Methods**, select 'Credit/Debit cards' and check the payer authentication checkbox on any credit card types you want to support Payer Authentication.

Upgrade to 3DS2.0

If you are currently using CYBS cartridge and would like to upgrade to 3DS2.0, please refer doc **CyberSource Storefront Reference Architecture LINK Cartridge 3DS2.docx** under cartridge documentation folder.

SCA (Strong Customer Authentication):

If response code 478 is received, the issuer is declining the authorization request but advising that if the card holder gets SCA, the issuer will approve the authorization.

Proposed Approach:

Automatically retry the transaction adding Payer Auth Mandate Challenge (code 04) to force cardholder SCA challenge. This flow should appear seamless to the card holder

Site Preferences:

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/meta/Cybersource_PayerAuthentication.xml"** inBusiness Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource_PayerAuthentication** and set values for the following parameters:

Field	***Description***
IsSCAEnabled	Enable Strong Customer Authentication

Table 7: Cybersource SCA Preferences

Set the value for IsSCAEnabled to yes to use Strong Customer Authentication feature.

5.2. Apple Pay

5.2.1. Create a merchant identifier in Apple Portal

A merchant identifier uniquely identifies you to Apple Pay as a merchant who can accept payments. You can use the same merchant identifier for multiple native and web apps. It neverexpires.

1. Go to Apple portal: https://help.apple.com/developer-account/#/devb2e62b839?sub=dev103e030bb
2. In Certificates, Identifiers & Profiles, select Identifiers from the sidebar, then click theAdd button (+) in the upper-left corner.
3. Select Merchant IDs, then click Continue.
4. Enter the merchant description and identifier name, then click Continue.
5. Review the settings, then click Register.

5.2.2. Enrolling in Apple Pay in Cybersource

To enroll in Apple Pay:

1. Log in to the Business Center:
    - Test: link
    - Production: link
2. On the left navigation pane, click the **Payment Configuration** icon.
3. Click **Digital Payment Solutions**. The Digital Payments page appears.
4. Click **Configure**. The Apple Pay Registration panel opens.
5. Enter your Apple Merchant ID. (Created in Step 1.4)
6. Click **Generate New CSR**.
7. To download your CSR, click the **Download** icon next to the key.
8. Follow your browser's instructions to save and open the file.

5.2.3. Complete the enrollment process by submitting your CSR to Apple

Create a payment processing certificate: A payment processing certificate is associated with your merchant identifier and used to encrypt payment information. The payment processing certificate expires every 25 months. If the certificate is revoked, you can recreate it.

1. In Certificates, Identifiers & Profiles, select Identifiers from the sidebar.
2. Under Identifiers, select Merchant IDs using the filter in the top-right.
3. On the right, select your merchant identifier. Note: If a banner appears at the top of thepage saying that you need to accept an agreement, click the Review Agreement button,and follow the instructions before continuing.
4. Under Apple Pay Payment Processing Certificate, click Create Certificate.
5. Click Choose File.
6. In the dialog that appears, select the CSR file downloaded from 5.2.2., then clickChoose.
7. Click Continue.

5.2.4. Configure Apple Pay in SFCC Business Manager

Business Manager Configuration

1. Go to: **"Merchant Tools > Site Preferences > Apple pay**
2. Check **"Apple Pay Enabled?"**
3. Fill in the "**Onboarding**" form:
   - Ensure "**Apple Merchant ID" and "Apple Merchant Name"** match settings in your Apple account
   - Ensure all other fields match your supported Cybersource settings
4. Fill in the "**Storefront Injection**" form:
   - Selects where Apple Pay buttons should be displayed on your site.
5. Fill in **"Payment Integration"** form:
   - Leave all form fields blank
   - Ensure **"Use Basic Authorization"** is checked
6. Click **"Submit"**

5.2.5. Domain Registration in SFCC Business Manager

1. Go to: **"Merchant Tools > Site Preferences > Apple Pay**
2. Under **Domain Registration** section
   a. Click on **Register Apple Sandbox** under Apple Sandbox section for registering SFCC to Apple Sandbox account.
   b. Click on **Register Apple Production** under Apple Production section for registering SFCC to Apple Production account.

5.2.6. Payment Processor

In the Business Manager, go to Merchant Tools > Ordering > Payment Methods andselect DW_APPLE_PAY. And in DW_APPLE_PAY details, double check if Payment Processor = "CYBERSOURCE_CREDIT"

5.2.7. Site Preferences

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/meta/ApplePay.xml"** in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > Apple Pay** and set values for the following parameters:

Field	***Description***
ApplePayTransactionType	Select Sale/Auth transaction type

Table 8: Cybersource ApplePay Preferences

5.3. PayPal

Cybersource Cartridge supports the following for PayPal:

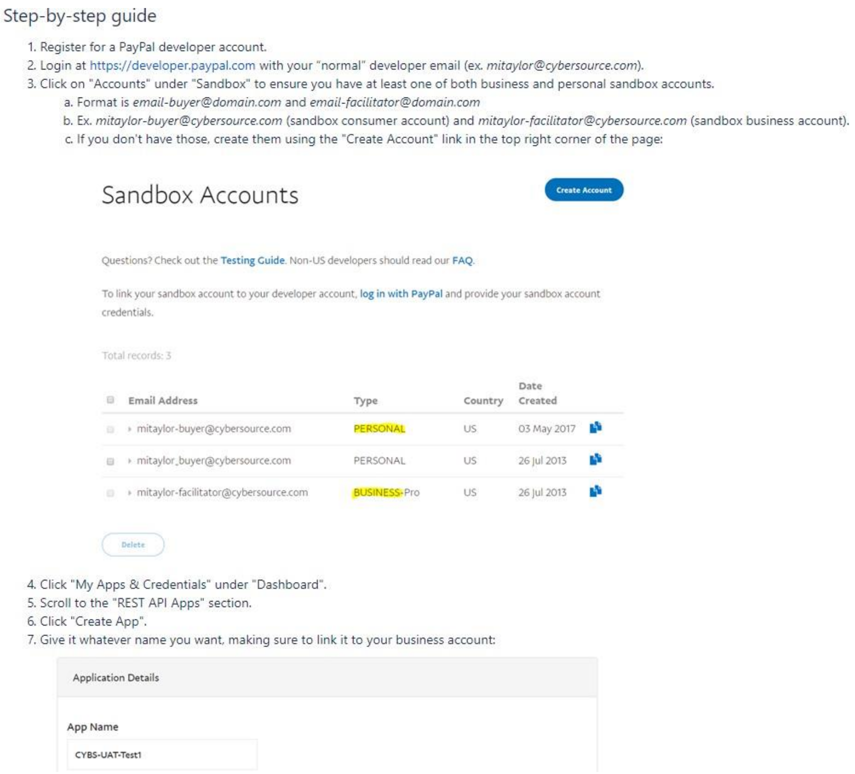
a. PayPal Express
b. PayPal Credit
c. PayPal Billing Agreement

Prerequisite

Prior to development phase, there are a generic set of configurations that a development team needs to account for. These configurations include:

1. PayPal developer account
2. PayPal sandbox account

Screenshot of the detailed set of configurations for #1 & #2.



Step-by-step guide

1. Register for a PayPal developer account.
2. Login at https://developer.paypal.com with your "normal" developer email (ex. *mitaylor@cybersource.com*).
3. Click on "Accounts" under "Sandbox" to ensure you have at least one of both business and personal sandbox accounts.
   a. Format is *email-buyer@domain.com* and *email-facilitator@domain.com*
   b. Ex. *mitaylor-buyer@cybersource.com* (sandbox consumer account) and *mitaylor-facilitator@cybersource.com* (sandbox business account).
   c. If you don't have those, create them using the "Create Account" link in the top right corner of the page:

Sandbox Accounts

Questions? Check out the Testing Guide. Non-US developers should read our FAQ.

To link your sandbox account to your developer account, log in with PayPal and provide your sandbox account credentials.

Total records: 3

□	Email Address	Type	Country	Date Created
□	mitaylor-buyer@cybersource.com	PERSONAL	US	03 May 2017
□	mitaylor_buyer@cybersource.com	PERSONAL	US	26 Jul 2013
□	mitaylor-facilitator@cybersource.com	BUSINESS-Pro	US	26 Jul 2013

4. Click "My Apps & Credentials" under "Dashboard".
5. Scroll to the "REST API Apps" section.
6. Click "Create App".
7. Give it whatever name you want, making sure to link it to your business account:

Application Details

App Name: CYBS-UAT-Test1

3. Linking developer and sandbox account. On creating a PayPal developer account, get in touch with the Cybersource team, share the developer account details and get the developers' details configured on Cybersource (BackOffice Configuration tool). Share the following keys with Cybersource:

- ClientID (**paypalgateway_client_key)** - Follow this link to generate.
- Secret (**paypalgateway_secret_phrase)** - Follow this link to generate.
- Merchant Account ID (**paypalgateway_mid) –** Follow this link to generate.
- Merchant email (**paypalgateway_merchant_email)** - Follow this link to generate.

5.3.1. To setup PayPal Express

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_Paypal.xml**" in Business Manager (**Administration > Site Development > Import & Export**).

Step 2: Go to **Merchant Tools** > **Site Preferences > Custom Preferences > Cybersource_Paypal** and set values for the parameter:

Field	***Description***
CsEnableExpressPaypal	Effectively enables or disables the PayPal Express checkout.
Paypal Order Type	The type of authorization to follow for PayPal orders. Select STANDARD for Authorize & Capture or select CUSTOM for just Authorize

Table 9: Paypal Express Preferences

5.3.2. To setup PayPal Credit

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_Paypal.xml**" in Business Manager (**Administration > Site Development > Import & Export**).

Step **Preferences > Custom Preferences > Cybersource_Paypal** and set values for the parameter:

Field	***Description***
Paypal Order Type	The type of authorization to follow for PayPal orders. Select STANDARD for Authorize & Capture or select CUSTOM for just Authorize
Paypal Address Override	If this set to true, then address of buyer will be override by the shipping address provided in shipping page

Table 10: Paypal Credit Preferences

5.3.3. To setup PayPal Billing Agreement

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_Paypal.xml**" in Business Manager (**Administration > Site Development > Import & Export**).

Step 2: Go to **Merchant Tools** > **Site Preferences > Custom Preferences > Cybersource_Paypal** and set values for the parameter:

Field	***Descri ption***
Billing Agreement	Effectively enables or disables the PayPal Billing Agreement.
Paypal Order Type	The type of authorization to follow for PayPal orders. Select STANDARD for Authorize & Capture or select CUSTOM for just Authorize

Table 11: Paypal Billing Preferences

5.3.4. To setup Decision Manager for PayPal

Refer to this link if you want to setup Decision Manager feature for PayPal transactions.

Go to **Merchant Tools > Site Preferences > Custom Preferences >** Cybersource: Decision Manager Configuration and set values for the parameter:

Field	***Description***
Decision Manager Enable for Paypal	Enable or disable Decision Manager for PayPal transactions.

Table 12: Decision Manager for Paypal

5.4. Google Pay

Prerequisite:

Before proceeding to BM configuration, a Merchant Account needs to be created with Google.Please follow the link to create the merchant account with google.

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource_GooglePay.xml**" in BusinessManager (**Administration > Site Development > Import & Export**).

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_GooglePay** and set values for the parameter:

Field	***Description***
enableGooglePay	Enable or Disable Google Pay Service
googlePayMerchantID	Cybersource Merchant account ID
googlePaygatewayMerchantId	Matching setting on Google Account
googlePayTransactionType	Select Auth/Sale transaction

Table 13: GooglePay Preferences

5.5. Visa Checkout

Prerequisite:

1. To complete the checkout process successfully create Visa Checkout profile (link) on Cybersource business center console under 'Account Management > Digital Payment Solutions > Profiles > Enable Visa Checkout'.
2. Click profile tab and add profile, configure all the mandatory settings, also use API Keyfrom Setting Tab.
3. Create Secret key from **'Account Management > Client Integration Management >Payment Configuration > Key Management'**. Click Generate key and select shared secret.

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_VisaCheckout.xml**" in Business Manager **(Administration > Site Development > Import & Export)**.

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_VisaCheckout** and set values for the parameter:

Field	***Description***
cybVisaSdkJsLibrary	Sandbox: https://sandbox-assets.secure.checkout.visa.com/checkout-widget/resources/js/integration/v1/sdk.js Live: https://assets.secure.checkout.visa.com/checkout-widget/resources/js/integration/v1/sdk.js
cybVisaTellMeMoreLinkActive	Indicate whether Tell Me More Link to be displayed with VISA button true (default) false
cybVisaButtonColor	The color of the Visa Checkout button. standard or neutral.
cybVisaButtonSize	The size of the Visa Checkout button
cybVisaButtonHeight	The height of the Visa Checkout button in pixels.
cybVisaButtonImgUrl	Sandbox: https://sandbox.secure.checkout.visa.com/wall et-services-web/xo/button.png Live: https://secure.checkout.visa.com/wallet-services-web/xo/button.png
cybVisaCardBrands	Brands associated with card art to be displayed
cybVisaButtonWidth	The width of the Visa Checkout button in pixels.
cybVisaThreeDSSuppressChallenge	Whether a Verified by Visa (VbV) consumer authentication prompt is suppressed for this transaction. If true, VbV authentication is performed only when it is possible to do so without the consumer prompt. true - Do not display a consumer prompt false - Allow a

	consumer prompt
cybVisaExternalProfileId	Profile created externally by a merchant whom Visa Checkout uses to populate settings
cybVisaSecretKey	The secret key specified Visa Checkout account profile
cybVisaAPIKey	The Visa Checkout account API key specified in Cybersource business center
cybVisaThreeDSActive	Whether Verified by Visa (VbV) is active for this transaction. If Verified by Visa is configured, you can use threeDSActive to deactivate it for the transaction; otherwise, VbV will be active if it has been configured
CybVisaButtonOnCart	Cybersource Visa Button display on minicart and cart
cybVisaTransactionType	Select Sale/Auth transaction type

Table 14: Visa Checkout Preferences

To setup Decision Manager for Visa Checkout

Refer to this link if you want to setup Decision Manager feature for ”Visa Checkout”transactions.

Go to **Merchant Tools > Site Preferences > Custom Preferences >** Cybersource: Decision Manager Configuration and set values for the parameter:

Field	***Description***
CS Decision Manager Enable	Enable or disable Decision Manager for Visa Checkout transactions.

Table 15: Decision Manager for Visa Checkout

5.6. Bank Transfer

Bank Transfer supports 3 types of Payment methods –

- SOFORT
- BANCONTACT
- IDEAL

Payment Method	Country	Services
Bancontact	Belgium	Sale Check Status

		Refund
iDEAL	Netherlands	Options Sale Check Status Refund
Sofort	Austria Belgium Germany Italy Netherlands Spain	Sale Check Status Refund

Table 16: Bank Transfer Service Support by Country

Bank transfer supports 4 different services:

- **Option Service:** This service is valid only for iDEAL transactions. The options service (apOptionsService) retrieves a list of bank option IDs and bank names which you can display to the customer on your web site
- **Sale Service:** The sale service (apSaleService) returns the redirect URL for customer's bank. The customer is directed to the URL to confirm their payment details.
- **Check Status Service:** The check status service returns the latest status of a transaction.It is a follow-on request that uses the request ID value returned from the sale service request. The request ID value links the check status request to the payment transaction.
- **Refund Service:** The refund service request (apRefundService) is a follow-on request that uses the request ID value returned from the sale request. The request ID value links the refund transaction to the original payment transaction.

Bank Transfer functionality is specific to PMs with sale and check status service.
SFRA makes the call to Cybersource Sale service to authorize the purchase amount. A secondary call is made to check a Status service to determine result of the authorization, andthe subsequent Order creation or failure.

Implementation
Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_BankTransfer.xml**" in BusinessManager (**Administration > Site Development > Import & Export**).

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_BankTransfer** and set values for the parameter:

Field	***Description***
Merchant Descriptor Postal Code(merchantDescriptorPostalCode)	Merchant Descriptor Postal Code
Merchant Descriptor(merchantDescriptor)	Merchant Descriptor
Merchant Descriptor Contact(merchantDescriptorContact)	Merchant Descriptor Contact

Merchant Descriptor State(merchantDescriptorState)	Merchant Descriptor State
Merchant Descriptor Street(merchantDescriptorStreet)	Merchant Descriptor Street
Merchant Descriptor City(merchantDescriptorCity)	Merchant Descriptor City
Merchant Descriptor Country(merchantDescriptorCountry)	Merchant Descriptor Country

Table 17: Bank Transfer Preferences

Step 3: Go to **Merchant Tools > Ordering > Payments Methods > BANK_TRANSFER** and setvalues for the parameter:

Field	***Description***
isBicEnabled	Attribute to check if BIC field is required for EPS and GIROPAY to display on billing page
isSupportedBankListRequired	Attribute to check if bank list is required for IDEAL to display on billing page

Table 18: Bank Transfer Preferences

To setup Decision Manager for Bank Transfer

Refer to this link if you want to setup Decision Manager feature for Bank Transfer transactions.
Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource: Decision Manager Configuration** and set values for the parameter:

Field	***Description***
Decision Manager Enable for Bank Transfer	Enable or disable Decision Manager for Bank Transfer transactions.

Table 19: Decision Manager for Bank Transfer

5.7. Alipay

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_Alipay.xml**" in Business Manager (**Administration > Site Development > Import & Export**).

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource_Alipay** and set values for the parameter:

Field	***Descripti on***
apPaymentType	Alipay Payment Type for Domestic as well as International Payment
apTestReconciliationID	Test Reconciliation ID for Alipay

Table 20: Alipay Preferences

Step 3: Under '**Merchant Tools > Ordering > Payment Methods**' Make sure the 'ALIPAY'payment method is enabled and configured to use the 'CYBERSOURCE_ALIPAY' paymentprocessor.

5.8. Klarna

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_Klarna.xml**" in Business Manager (**Administration > Site Development > Import & Export**).

Step 2: On the Payment Methods page, Select the locale (language) you want to set up, thenselect the Klarna payment method.

Step 3: Select the appropriate bill-to language setting under the 'Klarna' custom attribute Group.

Step 4 : Generate a p12 file by following the steps in doc .

Step 5: Navigate to Administration > Operations > Private Keys and Certificates and upload the file generated in step 4 (Add an alias and password (added while creating p12 file))

Step 6: Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource_Klarna** and set values for the parameter:

Field	Description
Klarna JS API Path	Klarna JS API Library Path
IsKlarnaPaymentFlowModeEnabled	Is Klarna PaymentFlowMode Enabled
klarnaPrivateKeyAlias	Private Key Alias of imported Key in Private Keys and Certificates
Klarna Decision Manager Required	Enable or Disable Decision Manager for Klarna transactions.

Table 21: Klarna Preferences

5.9. WeChat Pay

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_WeChat.xml**" in Business Manager (**Administration > Site Development > Import & Export**).

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource_Klarna** and set values for the parameter:

Field	**Description**

Test Reconciliation ID for WeChat Pay	Sets the status of the AP SALE such as settled, pending, abandoned, or failed
WeChatPayTransactionTimeout	Transaction Timeout for QR Code in WeChat Pay in seconds
CheckStatusServiceInterval	Interval in seconds before checking status of AP sale
NumofCheckStatusCalls	Max number of calls to check status for each AP sale

Table 22: WechatPay Preferences

6. Configure Features (OPTIONAL)

6.1. Tax Calculation

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in

"4.2: Upload metadata" orimport "**metadata/sfra_meta/meta/Cybersource_TaxConfiguration.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_TaxConfiguration** and set values for the parameter:

Field	***Description***
CS Tax Calculation Enabled	Enable or disable Cybersource tax service
CS Tax Calculation Nexus States List	List of states to charge tax in
CS Tax Calculation No Nexus States List	List of States to not charge tax in
CS Tax Calculation Default Product Tax Code	Default tax code used when tax code is not set on a product
CS Tax Calculation Purchase Order Acceptance City	Purchase order acceptance state code
CS Tax Calculation Purchase Order Acceptance Zip Code	Purchase order acceptance zip code
CS Tax Calculation Purchase Order Acceptance Country Code	Purchase order acceptance country code
CS Tax Calculation Purchase Order Origin City	Purchase order origin city
CS Tax Calculation Purchase Order Origin StateCode	Purchase order origin state code
CS Tax Calculation Purchase Order Origin ZipCode	Purchase order origin zip code
CS Tax Calculation Purchase Order Origin Country Code	Purchase order origin country code
CS Tax Calculation ShipFrom City	Ship from city
CS Tax Calculation ShipFrom StateCode	Ship from state code
CS Tax Calculation ShipFrom ZipCode	Ship from zip code

CS Tax Calculation ShipFrom Country Code	Ship from country code

Table 23: Cybersource Taxation Preferences

6.2. Delivery Address Verification

Step 1: To enable this service, Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource_DeliveryAddressVerification** and set the 'CS DAV Delivery Address Verification Enabled' preference to 'Yes'.

Step 2: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_DeliveryAddressVerification.xml**" in BusinessManager (**Administration > Site Development > Import & Export**)

Step 3: Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource_DeliveryAddressVerification** and set values for the parameter:

Field	***Description***
CS DAV Delivery Address Verification Enabled	Enable or disable the DAV service.
CS DAV Update Shipping Address with DAV Suggestion	Update the shipping address with the CS suggestion, if found.
CS DAV On Failure	Accept or Reject the order if DAV fails.

Table 24: Cybersource DAV Preferences

6.3. Address Verification Service (AVS)

Assuming you have implemented the Credit Card Authorization service, you are ready to use the AVS service.

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_DeliveryAddressVerification.xml**" in BusinessManager (Administration > Site Development > Import & Export)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource_DeliveryAddressVerification** and set values for the parameter:

Field	***Description***
CS AVS Ignore AVS Result	Effectively enables or disables the AVS service
CS AVS Decline Flags	Leave empty to follow CS default decline flag strategy Enter flags separated by commas to overwrite the default flag rules

Table 25: Cybersource DAV Preferences

6.4. Device Fingerprint

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" orimport "**metadata/sfra_meta/meta/Cybersource_DeviceFingerprint.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_DeviceFingerprint** and set values for the parameter:

Field	***Description***
CS Device Fingerprint Enabled	Enable or Disable the Device Fingerprint Service
CS Device Fingerprint Organization ID	Device Fingerprint Organization ID
CS Device Fingerprint ThreatMetrix URL	URL pointing to JS that generates and retrieves the fingerprint
CS Device Fingerprint Time To Live	Time, in milliseconds between generating a new fingerprint for any given customer session

Table 26: Cybersource Device Fingerprint Service

6.5. Decision Manager

Refer to this link to learn about Cybersource's Decision Manager.

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in

"4.2: Upload metadata" orimport "**metadata/sfra_meta/meta/Cybersource_DecisionManager.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_DecisionManager** and set values for the parameter:

Field	***Description***
CS Decision Manager Enabled	Enable or Disable Decision Manager for Credit card transactions.

Table 27: Cybersource Decision Manager Service

Step 3: SFRA storefront versions 3.2.0 or lower contain a hook that interfere with this service.While the hook manager has been updated in later versions of SFRA to prevent this, the CS cartridge is not yet compatible with those storefront versions. As suggested by SFCC, manual removal of the following hook from SFRA is required for this integration to function properly.

Remove

```
{
     "name": "app.fraud.detection",
     "script": "./cartridge/scripts/hooks/fraudDetection"
}
```

From app_storefront_base/hooks.json

Step 4: To enable **Decision Manager Order Update Job**: Decision Manager Order Update Job uses a REST API to retrieve order decisions from Cybersource and update the order confirmation status in SFCC.

To Integrate this job into your site, follow the below steps:

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import **"metadata/sfra_meta/jobs.xml"** in Business Manager (**Administration >Operations > Import & Export**)

Step 2: Navigate to '**Administration > Operations > Job**'. Select the Job '**CyberSource:Decision Manager Order Update**'.

Step 3: Select the "Job Steps" tab. Select the Sites you want the Job to run on, from the 'Scope' button.

Step 4: Select "UpdateOrderStatus" and update the following "custom parameters".

Field	***Description***
MerchantId	CS Merchant ID for the account to get Decisions from
SAFlexKeyID	Key ID. Work with CS to generate this value or Follow link to generate keys.
SAFlexSharedSecret	Shared secret. Work with CS to generate this value or Follow link to generate secret.

Table 28: Cybersource Decision Manager Job Update

Step 5: Navigate to the 'Schedule and History' tab and configure the frequency you would like the job to run.

Step 6: Ensure the 'Enabled' check box is selected.

Step 7: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_DecisionManager** set values for the parameter:

Field	***Site Pref Group***	***Description***
CS Decision Manager OrderUpdate Lookback time	Cybersource: Decision Manager	Number of hours the job will look back for new decisions. CS does not support lookbacks over 24 hours. Do not set above 24
Secure Acceptance Flex Host Name	Cybersource: Secure Acceptance	Host Name. CS can provide this value

Table 29: Cybersource Decision Manager Job

Step 8: When moving to a production environment, the URL for the API call needs to be updated. This can be done in:
Administration > Operations > Services > Service Credentials > ConversionDetailReport – Details

6.6. Payment Tokenization

Refer to this link to learn about Cybersource's Token Management service.

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_Tokenization.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_Tokenization** and set values for the parameter:

Field	***Description***
CS Tokenization Enabled	Enable or Disable the Tokenization Service
CS Subscription Tokens Enabled	Enable the request of a subscription token on credit card authorizations
LimitSavedCardRate	Enable Save Card Limit feature
SavedCardLimitFrame	Provide the number of cards that can be saved in a certain time period
SavedCardLimitTimeFrame	Provide the number of hours that saved card attempts are counted

Table 30: Cybersource Tokenization Service

6.7. Subscription Token Creation

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource_Tokenization.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences >Cybersource_Tokenization** and set values for the parameter:

Field	***Description***
CS Subscription Tokens Enabled	Enable or Disable the option to generate subscription tokens

Table 31: Cybersource Subscription Token Service

6.8. Capture Service

6.8.1. Credit Card

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make capture requests is in the form of a single function:

CCCaptureRequest(requestID, merchantRefCode, paymentType, purchaseTotal, currency)

This function can be found in the script 'scripts/facade/CardFacade.js'. A working example of how to use this function can be found in the CYBServicesTesting-CaptureService controller. You will first get an instance of the CardFacade object, and make the call as follows:

```
var CardFacade = require('~/cartridge/scripts/facade/CardFacade');
var serviceResponse = CardFacade.CCCaptureRequest(requestID, merchantRefCode,
paymentType, paymentTotal, currency);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes. For detailed explanations of all possible fields and values, reference the Officials Cybesource documentation for the CCCapture Service.

Step 4: Enter Capture Request Parameters:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. 'USD')

Table 32: Cybersource CC Capture Service

6.8.2. Visa Click To Pay

Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make capture requests is in the form of a single function: VCCaptureRequest(requestID, merchantRefCode, paymentType, purchaseTotal, currency, orderid)

This function can be found in the script 'scripts/visacheckout/facade/VisaCheckoutFacade.js'. A working example of how to use this function can be found in the CYBServicesTesting-CaptureService controller. Youwill first get an instance of the VisaCheckoutFacade object, and make the call as follows:

```
var VisaCheckoutFacade =
require('~/cartridge/scripts/visacheckout/facade/VisaCheckoutFacade');
var serviceResponse = VisaCheckoutFacade.VCCaptureRequest(requestID,
merchantRefCode, paymentType, paymentTotal, currency, orderid);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Capture Request Parameters:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. ‘USD’)
orderid	SFCC Order Number

Table 33: Cybersource Visa Checkout Capture Service

6.8.3. PayPal

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make capture requests is in the form of a single function: PayPalCaptureService(requestID, merchantRefCode, paymentType, purchaseTotal, currency)

This function can be found in the script ‘scripts/paypal/facade/PayPalFacade.js’. A working example of how to use this function can be found in the CYBServicesTesting-CaptureService controller. You will first get an instance of the PayPalFacade object, and make the call as follows:

```
var PayPalFacade = require('~/cartridge/scripts/paypal/facade/PayPalFacade');
var serviceResponse = PayPalFacade.PayPalCaptureService(requestID, merchantRefCode,
paymentType, paymentTotal, currency);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Capture Request Parameters:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number

paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. ‘USD’)

Table 34: Cybersource PaypalCapture Service

6.8.4. Google Pay

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make capture requests is in the form of a single function: GPCaptureRequest(requestID, merchantRefCode, paymentType, purchaseTotal, currency)

This function can be found in the script ‘scripts/mobilepayments/facade/MobilePaymentFacade.js’. A working example of how to use this function can be found in the CYBServicesTesting-CaptureService controller. You will first get an instance of the MobileCheckoutFacade object, and make the call as follows:

```
var MobileCheckoutFacade =
require('~/cartridge/scripts/mobilepayments/facade/MobilePaymentFacade');
var serviceResponse = MobileCheckoutFacade.GPCaptureRequest(requestID,
merchantRefCode, paymentType, paymentTotal, currency);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Capture Request Parameters:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. ‘USD’)

Table 35: Cybersource Googlepay Capture Service

6.9. Auth Reversal Service

6.9.1. Credit Card

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager

(**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make auth reversal requests is in the form of a single function: CCAuthReversalService(requestID, merchantRefCode, paymentType, currency, amount)

This function can be found in the script 'scripts/facade/CardFacade.js'. A working example of how to use this function can be found in the CYBServicesTesting-CCAuthReversalService controller. You will first get an instance of the CardFacade object, and make the call as follows:

```
var CardFacade = require('~/cartridge/scripts/facade/CardFacade');
var serviceResponse = CardFacade.CCAuthReversalService (requestID,
merchantRefCode, paymentType, currency, amount);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.For detailed explanations of all possible fields and values, reference the Official Cybersource documentation for the CCAuthReversal Service.

Step 4: Enter Authorization Reversal Request Parameter:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. 'USD')

Table 36: Cybersource CC Auth Reversal Service

6.9.2. Visa Click To Pay

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make auth reversal requests is in the form of a singlefunction: VCAuthReversalService(requestID, merchantRefCode, paymentType, currency, amount, orderid)

This function can be found in the script 'scripts/visacheckout/facade/VisaCheckoutFacade.js'. A working example of how to use this function can be found in the CYBServicesTesting-CCAuthReversalService controller. You will first get an instance of the VisaCheckoutFacade object, and make the call as follows:

```
var VisaCheckoutFacade =
require('~/cartridge/scripts/visacheckout/facade/VisaCheckoutFacade');
var s serviceResponse = VisaCheckoutFacade.VCAuthReversalService(requestID,
merchantRefCode, paymentType, currency, amount, orderid);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Authorization Reversal Request Parameter:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. 'USD')
orderid	SFCC Order Number

Table 37: Cybersource VisaCheckout Auth Reversal Service

6.9.3 PayPal

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make auth reversal requests is in the form of a singlefunction: PayPalReversalService(requestID, merchantRefCode, paymentType, purchaseTotal, currency)

This function can be found in the script 'scripts/paypal/facade/PayPalFacade.js'. A working example of how to use this function can be found in the CYBServicesTesting-CCAuthReversalService controller. You will first get an instance of the PayPalFacade object, and make the call as follows:

```
var PayPalFacade =
require('~/cartridge/scripts/paypal/facade/PayPalFacade');
var serviceResponse = PayPalFacade.PayPalReversalService(requestID,
merchantRefCode, paymentType, amount, currency);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Authorization Reversal Request Parameter:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. ‘USD’)

Table 38: Cybersource Paypal Auth Reversal Service

6.9.4. Google Pay

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make auth reversal requests is in the form of a singlefunction: GPAuthReversalService(requestID, merchantRefCode, paymentType, currency, amount)

This function can be found in the script ‘scripts/mobilepayments/facade/MobilePaymentFacade.js’. A working example of how to use this function can be found in the CYBServicesTesting-CCAuthReversalService controller. You will first get an instance of the MobileCheckoutFacade object, and make the call as follows:

```
var MobileCheckoutFacade =
require('~/cartridge/scripts/mobilepayments/facade/MobilePaymentFacade');
var serviceResponse = MobileCheckoutFacade.GPAuthReversalService(requestID,
merchantRefCode, paymentType, currency, amount);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Authorization Reversal Request Parameter:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total

currency	Currency code (ex. ‘USD’)

Table 39: Cybersource Googlepay Auth Reversal Service

6.10. Credit Service

6.10.1. Credit Card

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make credit requests is in the form of a single function: CCCreditRequest(requestID, merchantRefCode, paymentType, purchaseTotal, currency)

This function can be found in the script ‘scripts/facade/CardFacade.js’. A working example of how to use this function can be found in the CYBServicesTesting-CreditService controller. Youwill first get an instance of the CardFacade object, and make the call as follows:

```
var CardFacade = require('~/cartridge/scripts/facade/CardFacade');
var serviceResponse = CardFacade.CCCreditRequest(requestID, merchantRefCode,
paymentType, paymentTotal, currency);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.For detailed explanations of all possible fields and values, reference the Official Cybersource documentation for the CCCredit Service.

Step 4: Enter Credit Request Parameters:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. ‘USD’)

Table 40: Cybersource CC Refund Service

6.10.2. Visa Click to Pay

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter

CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make credit requests is in the form of a single function: VCCreditRequest(requestID, merchantRefCode, paymentType, purchaseTotal, currency, orderid)

This function can be found in the script ‘scripts/visacheckout/facade/VisaCheckoutFacade.js’. A working example ofhow to use this function can be found in the CYBServicesTesting-CreditService controller. You will first get an instance of the VisaCheckoutFacade object, and make the call as follows:

```
var VisaCheckoutFacade =
require('~/cartridge/scripts/visacheckout/facade/VisaCheckoutFacade');
serviceResponse = VisaCheckoutFacade.VCCreditRequest(requestID,
merchantRefCode, paymentType, paymentTotal, currency, orderid);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Credit Request Parameters:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. ‘USD’)
orderid	SFCC Order Number

Table 41: Cybersource Visachekout Refund Service

6.10.3. PayPal

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make credit requests is in the form of a single function: PayPalRefundService(requestID, merchantRefCode, paymentType, amount, currency)

This function can be found in the script ‘scripts/paypal/facade/PayPalFacade.js’. A working example of how to use this function can be found in the CYBServicesTesting-CreditService controller. You will first get an instance of the PayPalFacade object, and make the call as follows:

```
var PayPalFacade = require('~/cartridge/scripts/paypal/facade/PayPalFacade');
var serviceResponse = PayPalFacade.PayPalRefundService(requestID,
merchantRefCode, paymentType, paymentTotal, currency);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes. For detailed explanations of all possible fields and values, reference the Official Cybersource documentation for the CCCredit Service.

Step 4: Enter Credit Request Parameters:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. ‘USD’)

Table 42: Cybersource Paypal Refund Service

6.10.4. Google Pay

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make credit requests is in the form of a single function: GPCreditRequest(requestID, merchantRefCode, paymentType, purchaseTotal, currency)

This function can be found in the script ‘scripts/mobilepayments/facade/MobilePaymentFacade.js’. A working example of how to use this function can be found in the CYBServicesTesting-CreditService controller. You will first get an instance of the MobileCheckoutFacade object, and make the call as follows:

```
var MobileCheckoutFacade =
require('~/cartridge/scripts/mobilepayments/facade/MobilePaymentFacade');
serviceResponse = MobileCheckoutFacade.GPCreditRequest(requestID,
merchantRefCode, paymentType, paymentTotal, currency, orderid);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Credit Request Parameters:

Parameter Name	***Description***

requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. ‘USD’)
orderid	SFCC Order Number

Table 43: Cybersource Googlepay Refund Service

6.10.5. BankTransfer

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in “4.2: Upload metadata” or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make credit requests is in the form of a single function: BanktransferRefundService(requestID, merchantRefCode, paymentType, amount, currency)

This function can be found in the script ‘scripts/banktransfer/facade/BankTransferFacade.js. A working example ofhow to use this function can be found in the CYBServicesTesting-CreditService controller. You will first get an instance of the BankTransferFacade object, and make the call as follows:

```
var BanktransferFacade =
require('~/cartridge/scripts/banktransfer/facade/BankTransferFacade');
var serviceResponse = BanktransferFacade.BanktransferRefundService(requestID,
merchantRefCode, paymentType, paymentTotal, currency);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Credit Request Parameters:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. ‘USD’)

Table 44: Cybersource Banktransfer Refund Service

6.10.6. Alipay

Step 1: Upload Cybersource metadata in Business Manager. Otherwise follow the steps mentioned in "4.2: Upload metadata" or import "**metadata/sfra_meta/meta/Cybersource.xml**" in Business Manager (**Administration > Site Development > Import & Export**)

Step 2: Go to **Merchant Tools > Site Preferences > Custom Preferences > CyberSource** and Enter CyberSource Merchant ID, CyberSource Merchant Key values.

Step 3: The interface you will use to make credit requests is in the form of a single function: AliPayRefundService(requestID, merchantRefCode, paymentType, amount, currency)

This function can be found in the script 'scripts/alipay/facade/AlipayFacade.js'. A working example ofhow to use this function can be found in the CYBServicesTesting-CreditService controller. You will first get an instance of the AliPayFacade object, and make the call as follows:

```
var AliPayFacade = require('~/cartridge/scripts/alipay/facade/AlipayFacade');
var serviceResponse = AliPayFacade.AliPayRefundService(requestID,
merchantRefCode, paymentType, paymentTotal, currency);
```

The resulting serviceResponse object will contain the full response object generated by the request. The contents of this object will determine your logic in handling errors and successes.

Step 4: Enter Credit Request Parameters:

Parameter Name	***Description***
requestID	Transaction ID obtained from the initial Authorization
merchantRefCode	SFCC Order Number
paymentType	Payment Type used for the Authorization
purchaseTotal	Order Total
currency	Currency code (ex. 'USD')

Table 45: Cybersource Alipay Refund Service

6.11. Request Customizations

Step 1: To customize request objects, register the hook 'app.cybersource.modifyrequest' in your cartridges 'hooks.json' file. An example would look like this, replacing the script path with your own script:

```
{
      "name": "app.cybersource.modifyrequest",
      "script": "./cartridge/scripts/hooks/modifyRequestExample"
}
```

You can copy the 'scripts/hooks/modifyRequestExample' script from this cartridge into your own to use as a template for extending and modifying service request objects. Note, every hookmust return a valid request object for the given service. It is recommended that you reference the Cybersource documentation for details on the exact nature of any fields you wish to customize or add.

Step 2: The following hooks are available for you to define in this file:Modify Request hooks

Hook Name	***Service Request to modify***
CCAuth	Credit Card Authorization
PayerAuthEnroll	Payer Authentication Enrollment
PayerAuthValidation	Payer Authentication Validation
AuthReversal	Credit Card Authorization Reversal
Capture	Credit Card Capture
Credit	Credit Card Credit/Refund
Tax	Tax Calculation

Table 46: Cybersource Service

6.12. Supported Locales

Out of box cartridge supports most of the locales like English (United States), English (United Kingdom), French (FRANCE), English (Austria), German (GERMANY), Dutch (NETHERLANDS) andmore.

7. Test and Go Live

Test your integration, and configure your live account, so you can start processing live transactions.

Test your integration

Before you start accepting payments, test your integration in Test sandbox:

The sandbox simulates the live payment gateway. The sandbox never processes an actual payment. We do not submit sandbox transactions to financial institutions for processing. The sandbox environment is completely separate from the production environment, and it requires separate credentials. If you use your production credentials in the sandbox or vice versa, you get a 401 HTTP error.

Sign up for a sandbox account if you have not yet.

Use our test card numbers to make test payments: > The following test credit card numbers work only in the sandbox. If no expiration date is provided, use any expiration date after today’s date. If the card verification code is required and is not provided, use any 3-digit combination for Visa, Mastercard, Discover, Diners Club and JCB; use a 4-digit combination forAmerican Express.

Register SFCC sandbox to Apple Sandbox Account.

1. Go to **“Merchant Tools > Site Preferences > Apple pay**
2. Under **Domain Registration** section
   a. Click on **Register Apple Sandbox** under Apple Sandbox section for registering SFCC to Apple Sandbox account.

To manage your evaluation account, log in to the Test Business Center and do the following: -View test transactions. - Access administrator users and access privileges. - Create roles with predefined access permissions. - View reports.

Important Cybersource recommends that you submit all banking information and required integration services in advance of going live. Doing so will speed up your merchant account configuration.

Configure your live/production account

Once you have the credentials for the live/production environment:

1. Configure cartridge using your live/production account settings.
2. Test your integration

8. Upgrade Steps

Upgrade from 24.1.2 to 24.1.3:

Please follow the below steps to upgrade your cartridge version from 24.1.1 to 24.1.2.

1. Cartridge version 24.1.2 supports SFRA v7.0 and jQuery v3.7.1.
2. Cybersource's Storefront Reference Architecture Cartridge can be installed from Salesforce Commerce Cloud's marketplace link.
3. Create a p12 file that you can use to authenticate requests. Place the file generated in the webreferences2 folder in the same cartridge as the WSDL file and the file extension must be jks or pkcs12. Duplicate the CyberSourceTransaction.wsdl file, CyberSourceTransaction.wsdl.properties file and rename them with the same name as your respective p12 files.

   Repeat the above step to use multiple p12 files as per requirements.

Note: If you are currently using cartridge v21.1.0 or earlier or not upgradind to our cartridge version 24.1.3 can refer to **Merchant documentation guide for SFCC Ent P12 authentication.pdf** under cartridge documentation folder to be compliant with p12 update.

4. Upload metadata

   -> Cybersource cartridge installed from App Exchange comes with metadata to import
   -> Go to **Cybersource/metadata/sfra_meta/sites/** folder.
   -> Zip **sfra_meta** folder.
   -> Go to **Administration->Site Development->Site Import & Export** and upload **sfra_meta.zip** file.
   -> Import the uploaded zip file.
5. Navigate to Merchant Tools > Site Preferences > Custom Preferences > Cybersource and set values for the following parameters

Name	Description
CsP12_Name	Name of the p12 file added in webreferences2 folder in Step 3.2
CsP12_UserName	Friendly name extracted from p12 (run openssl pkcs12 -in CyberSourceTransaction.pkcs12 -info in terminal to extract friendly name of p12 key uploaded in webreferences2 folder.)
CsP12_Password	Enter the password added while generating p12 key.

6. Navigate to Administration > Operations > Private Keys and Certificates and upload the p12 file created in step 4 and provide alias and password (generated while creating the p12 file).

7. Navigate to Merchant Tools > Site Preferences > Custom Preferences > Cybersource_Klarna and set values for the following parameters for Klarna.

klarnaPrivateKeyAlias	Private Key Alias of imported Key in Private Keys and Certificates

Steps to upgrade Microform v0.11 to v2

8. In Business Manager, go to Administration > Operations > Services and click on the 'CybersourceFlexToken' Credentials. Ensure the appropriate URL is set for the environment you are configuring.
   Test: https://apitest.cybersource.com/microform/v2/sessions?format=JWT
   Production: https://api.cybersource.com/microform/v2/sessions?format=JWT

9. Navigate to Administration > Operations > Services and click on the 'AssymentricKeyManagement' Credentials. Ensure the appropriate URL is set for the environment you are configuring:
   Test: https://apitest.cybersource.com/flex/v2/public-keys
   Production: https://api.cybersource.com/flex/v2/public-keys

10. Go to **Merchant Tools > Site Preferences > Custom Preferences > Cybersource_SecureAcceptance** and set values for the parameter:

Field	**Description**	**Value to Set**
SA_Flex_AllowedCardNetworks	Configure card types for Cybersource Microform	VISA MASTER DISCOVER DINERSCLUB JCB MAESTRO AMEX ELO CUP JCREW CARTESBANCAIRES

Note: Please refer to steps added in **Cybersource for Salesforce B2C Commerce SOAP - Microform v2 Upgrade Steps.pdf** in our documentation section to update the version of Microform without updating to our cartridge version 24.1.3 which adds Microform v2.

9. Release Notes

Version 24.1.3 (February, 2025)

Enhancement:

- Added support to use multiple p12 files for p12 authentication.
- Upgraded Microform v0.11 to v2.
- Replaced ~ with * in require statements.

Bug Fixes:

- Fixed re-calculation of basket in handling 478.
- Added check for subscription id in TMS for SA_Redirect and SA_Iframe payment methods.
- Added a check for DDC url before form submission.

Version 24.1.2 (December, 2024)

- Upgraded jQuery v3.5.1 to v3.7.1.
- Addressed security findings.
- Bug Fix: Removed security code input field from bank transfer payment methods when CsSAType is set to SA_FLEX.
- Changed the authentication mechanism for the SOAP Toolkit to p12 authentication.

Version 24.1.1 (July, 2024)

- Upgraded the cartridge to support SFRA v7.0.
- Fixed card duplication in tokenization flow.

Version 24.1.0 (March, 2024)

- Implemented Direct API for Payer Authentication with Payer Auth Setup and Device Data Collection.
- Included the 3DS mandatory fields which were missing.
- Implemented SCA and handling 478 reason code.
- Implemented subscription creation in authorization call.

Version 23.1.2 (November, 2023)

- Updated the flex microform from v0.11.0 to v0.11.
- Updated API header in Http Signature Authentication.
- Bug fix for release 23.1.1

Version 23.1.1 (July, 2023)

- Upgraded the flex microform from v0.4.0 to v0.11.0

Version 23.1.0 (April 19, 2023)

- Implemented Sale for Credit Card, Google Pay, Visa Checkout and Apple Pay
- Extended the fix for other address fields in PayPal where non-english characters were not returned in the Cybersource response.
- Fixed incorrect unit product price in line items in Business center.

Version 22.1.3 (February 10, 2023)

- Upgraded the cartridge to support SFRA v6.3.
- Mapped requestID to transactionID in Klarna payment method.

- Added PayPal Pay Later message in the PayPal widget.

Version 22.1.2 (September 2, 2022)

- Updated the file name songbird.isml to songBird.isml
- Updated the cartridge to make it compatible with Salesforce B2C Commerce release 22.7
- Fixed the issue where payment details were updated with undefined.

Version 22.1.1 (July 14, 2022)

- Fixed an issue in PayPal where non-english characters was not returned in the Cybersource response.
- Added bin detection in 3ds flow.

Version 22.1.0 (May 15, 2022)

- Added paymentFlowMode configuration for Klarna payment.
- Added support for SCA changes for Irish processor.
- Updated the payment method Id for PayPal Credit to support PayPal payment for Salesforce Order Management.
- Added Transaction Type and Transaction Id for PayPal payment to support PayPal payment for Salesforce Order Management.
- Fixed a Decision Manager issue for Visa Checkout.
- Fixed a Decision Manager issue for Credit card transactions.
- Upgraded the Cybersource WSDL version to 1.192.

Version 21.3.0 (Nov 30, 2021)

- Disable Giropay and EPS Bank transfer method.
- Implemented Decision Manager service for Bank Transfer.
- Implemented Decision Manager service for Visa Checkout.
- Implemented Decision Manager service for PayPal.
- Added missing isSubscription custom field in metadata file. (GitHub issue#10)
- Fixed GitHub issue #71
- Fixed GitHub issue #69
- Fixed GitHub issue #75
- Implemented a new flag to override shipping address of PayPal.

Version 21.2.0 (Aug 27, 2021)

- Updated credit card form page in the My Account page with Flex Microform v0.11 implementation.
- Updated the cartridge to make it compatible with Salesforce B2C Commerce release 21.2.
- Implemented Decision manager in Payer Authentication call.
- Implemented standalone Decision manager service so merchants can call this service on demand.
- Implemented standalone service for Capture, Credit, Auth reversal for Klarna.
- Improved the quality of the code by linting.

Version 21.1.0 (Feb 15, 2021)

- Provide an option to enable/disable cartridge via business manager configuration.
- Upgrade the cartridge to support SFRA v5.1.0.

- Improved the metadata import process, now merchants can easily import all the metadata in a single go without having to import individual files.
- Improved the grouping of Site Preferences. All the related configurations can be found in the appropriate grouping.
- Storefront’s order confirmation email will now be sent to the storefront's registered email rather than to PayPal’s registered email.
- Made Payer authentication's site preference optional if you are not using PA. (GitHub issue#44)
- Improved the quality of the code by linting.
- Removed hard coding of environment variable for Google Pay. (GitHub issue#53)
- Removed hard coding of host URL from Decision manager job. (GitHub issue#49)
- Fixed the javascript method in songbird.isml. (GitHub issue#47)
- Removed server.replace from the cartridge. (GitHub issue#45)
- Improved the documentation of the cartridge.
- Converted .ds files to .js files.
- Replaced em or rems with px.
- Removed inline styles and separate them out into css files.
- Replaced importpackage() with require().
- Replaced the deprecated webreference package with webreference2 package.

Version 19.5.3 (Dec 30, 2020)

- Support 3ds for French Processor
- Remove flash code from Device fingerprint

Version 19.5.2 (Dec 04, 2020)

- Fix order token issue for SA_Redirect. (GitHub issue#57)
- Fix skip_decision_manager flag issue for SA_Silent Order Post (GitHub issue#46)

Version 19.5.1 (Nov 11, 2020)

- Improved security on accessing and modifying sensitive fulfillment-related actions on an order (e.g., order acceptance, canceling etc.).
- Add billing address to pa_enroll and pa_validate.
- Fix 3DS issues

Version 19.5.0 (May 20, 2020)

- Add WeChat Pay payment method.

Version 19.4.1 (March 17, 2020)

- Rate Limiting added to the My Accounts page, so a Merchant can determine the number of cards that can be edited or added.
- Update Cybersource WSDL to support Apache CXF v3 upgrade.

Version 19.4.0 (Feb 25, 2020)

- Updated cartridge to make it compatible with 4.3.0 version of SFRA.
- Bug fix on silent Post Payment method.
- Change reference code from “test” to unique ID in subscription creation and deletion
- call.

- Bug fix on JCB and Dinner club card processing.
- Minor fixes on Tax calculation that uses Cybersource tax services.
- Improved error handling on SA Flex and SA iframe.
- Bug fixes on PayPal button logo in Germany locale.
- Restructured cartridge folder to adhere to Salesforce’s new standards.

Version 19.3.4 (Nov 14, 2019)

- Supports Klarna payment and replace Conversion Detail Report to REST API

Version 19.3.2 (Sept 13, 2019)

- Bug fix on basic credit transactions to work as when Cardinal Cruise/Payer Auth is not configured

Version 19.3.1 (Sept 10, 2019)

- 3DS Documentation update

Version 19.3.0 (July 26, 2019)

- Update 3DS to version 2.0 utilizing Cardinal Cruise.

Version 19.1.1 (June 06, 2019)

- PayPal will now utilize the ‘CyberSource Endpoint’ site preference.
- Update to accommodate hitting the back button during SA redirect checkout.
- Added details to documentation.

Version 19.1.0 (Feb 08, 2019)

- Adds PayPal and PayPal Express Integrations
- Security patch to Test suite in Controllers Cartridge.
- Update to SA Flex date generation to require an en_US local.
- Fix to SA Flex payment response data being saved properly.

Version 18.1.2 (Jan 09, 2019)

- Update documentation with SFRA compatibility note.
- Remove public facing endpoints that can finalize an Order.
- Fix to SA SilentPost not processing Orders with Fraud status of 'Review'
- Ensure all Orders with Fraud decisions of 'Review' are placed in a 'Not Confirmed' state.
- Utilize CS endpoint preference to set Test or Production url endpoints.
- Always Send Date to Flex Token API in US English format.

Version 18.1.1 (Nov 06, 2018)

- Adds hooks to customize request objects.
- Separates subscription creation option to use a new site preference.
- Adds Facade for ‘Credit Card Credit’ Service.
- Adds Facade for ‘Credit Card Capture Service.
- Adds Facade for ‘Credit Card Auth Reversal’ Service.

Version 18.1.0 (Oct 25, 2018)

- Initial SFRA release.