SFCC FRAUD DETECTION (O-PLUX) SERVICE

CARTRIDGE INSTALLATION MANUAL

Table of Contents

1. Introduction

This document describes how to install and configure the fraud detection services cartridge in a Salesforce Commerce Cloud environment (referred below as SFCC). The cartridge was tested against 6.0.0 of SFRA and compatibility mode 21.7.

2. Components Overview

2.1 Functionalities Overview

This cartridge provides a fraud detection service operated by Cacco Inc.

2.2 Limits and Restrictions

- Target

  This cartridge can be used with either Storefront Reference Architecture (SFRA) .

- Language/Country restrictions

  The cartridge is developed for Japanese. It might not be working properly if you chose other language (Locale).

3. Integration Guide

3.1 Setup

3.1.1 Cartridge for a Storefront

- int_oplux
- int_oplux_sfra_changes

3.1.2 Cartridge for a Business Manager

- bm_oplux

3.1.3 Metadata for Integration

- system-objecttype-extensions.xml
- services.xml
- jobs.xml
- library.xml

The file for site import has the following structure. Compress the top folder into a zip file for use. In doing so, replace the folder name of [RefArchSharedLibrary] with the ID of the content library.

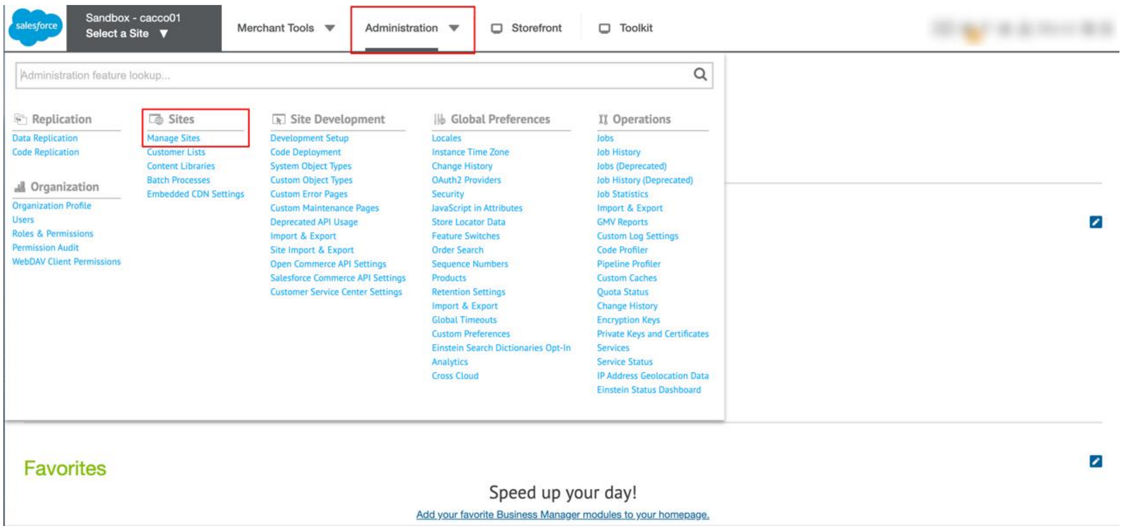
Top folder (site-template)

```
├─libraries
    └─RefArchSharedLibrary
        └─library.xml
├─services.xml
├─jobs.xml
└─meta
    └─system-objecttype-extensions.xml
```

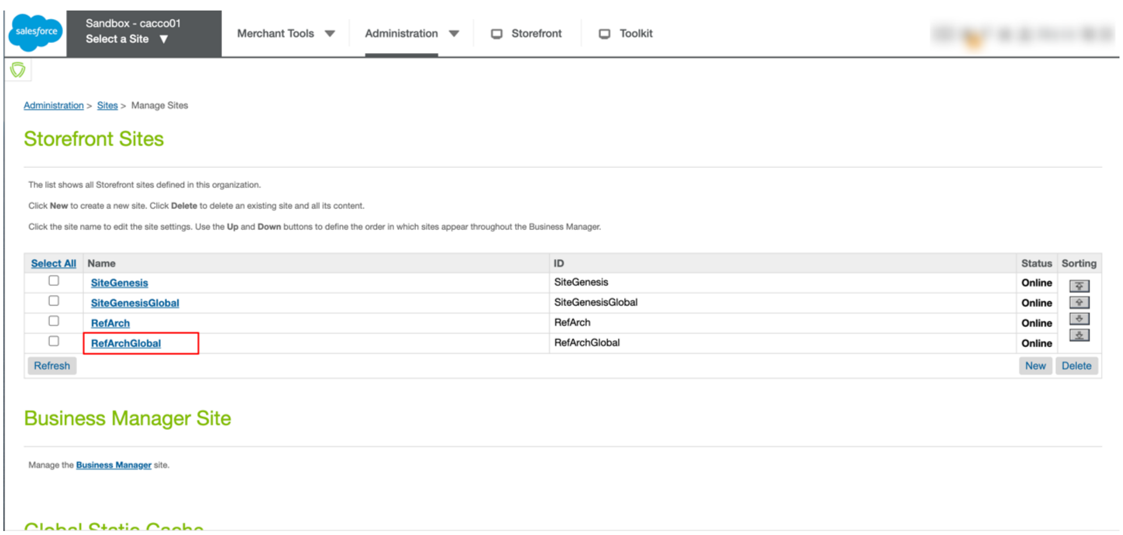
3.2 Components

3.2.1 Assigning a Cartridge to the Site

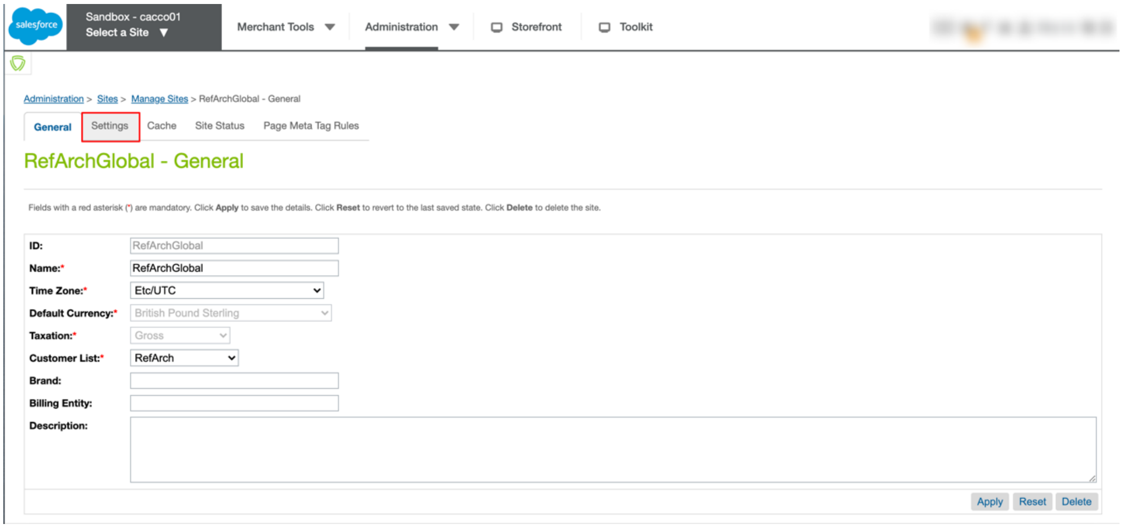
In Business Manager, go to **Administration > Sites > Manage Sites.**



Click on the site to manage.

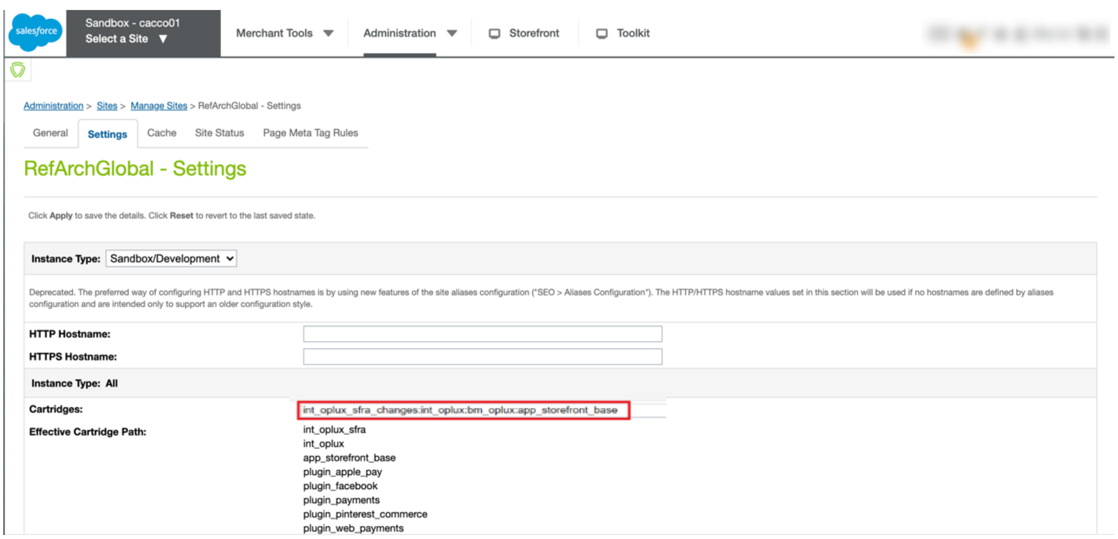


Open the “**Setting**” tab for Cartridge setting.



Set the cartridge path settings on the “**Settings**” tab as below.

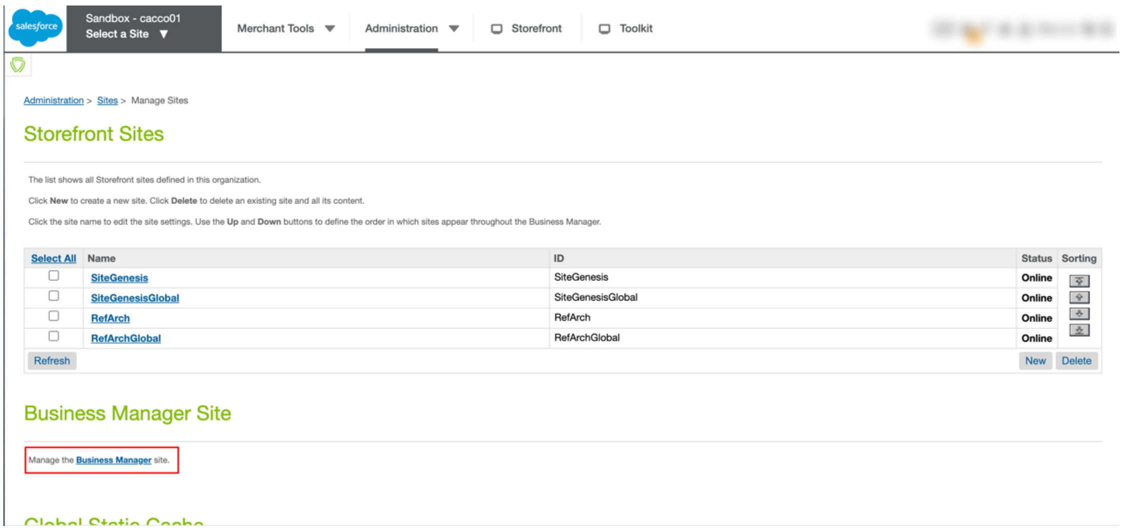
int_oplux_sfra_chages:int_oplux:bm_oplux:app_storefront_base



3.2.2 Admin Screen Configuration

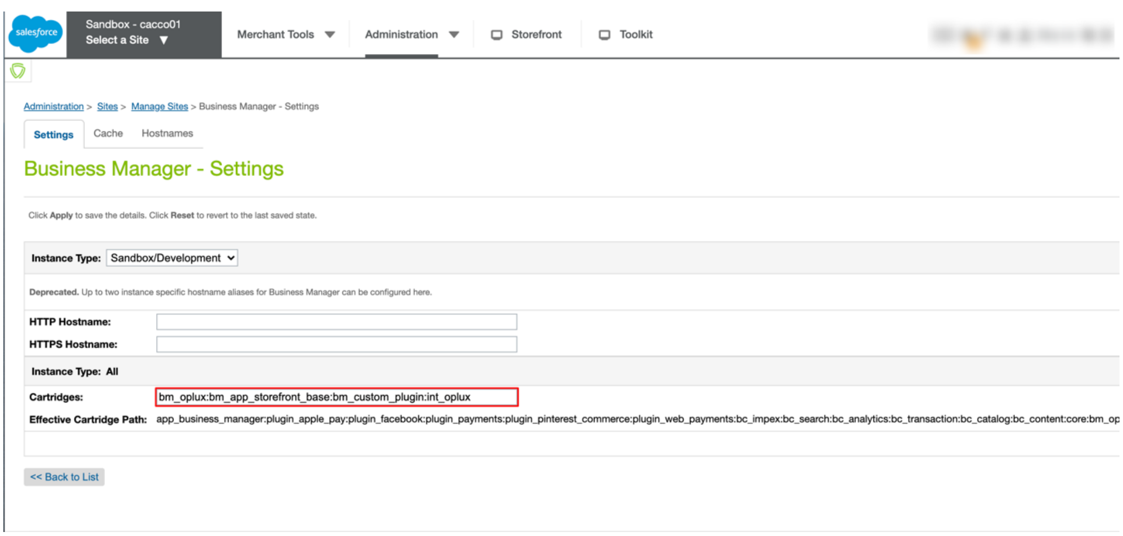
3.2.2.1 Setup Cartridges

Click the "**Back to List**" button, then go to "**Business Manager Site**" > "**Manage the Business Manager site**" link.



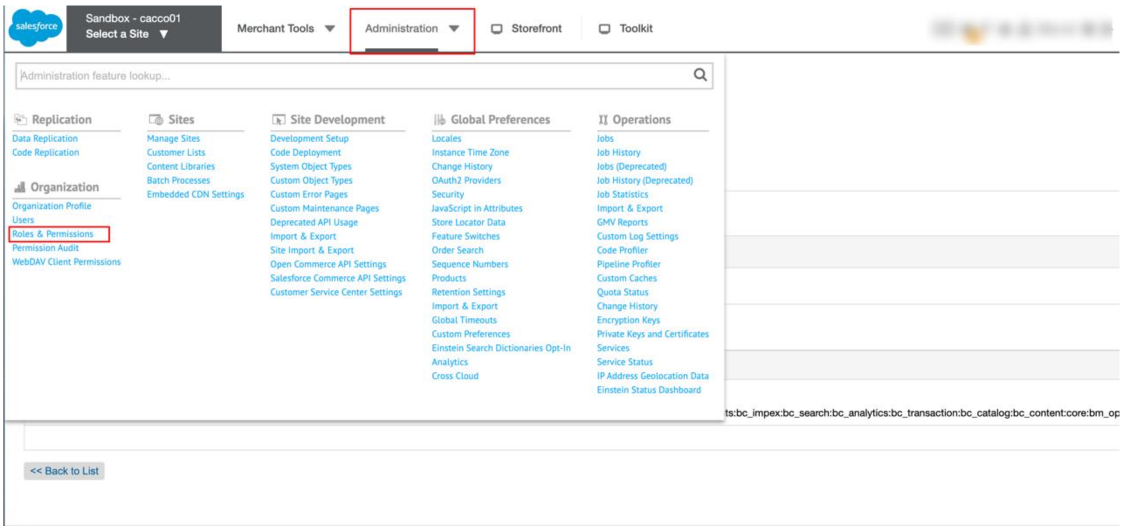
Set the cartridge path settings on the "**Settings**" tab as below.

bm_oplux:bm_app_storefront_base:bm_custom_plugin:int_oplux

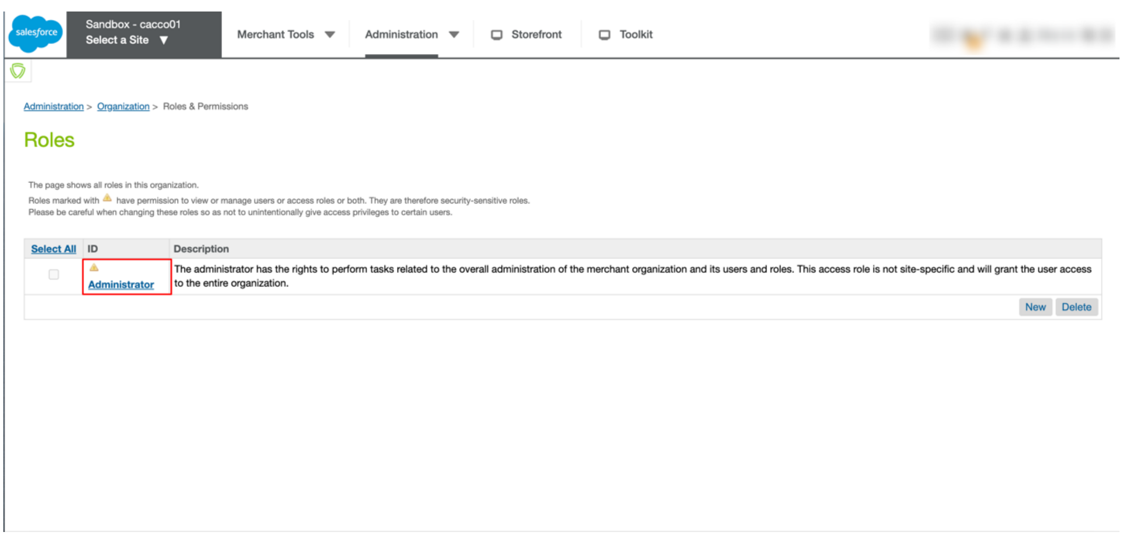


3.2.2.2 Admin Screen Configuration

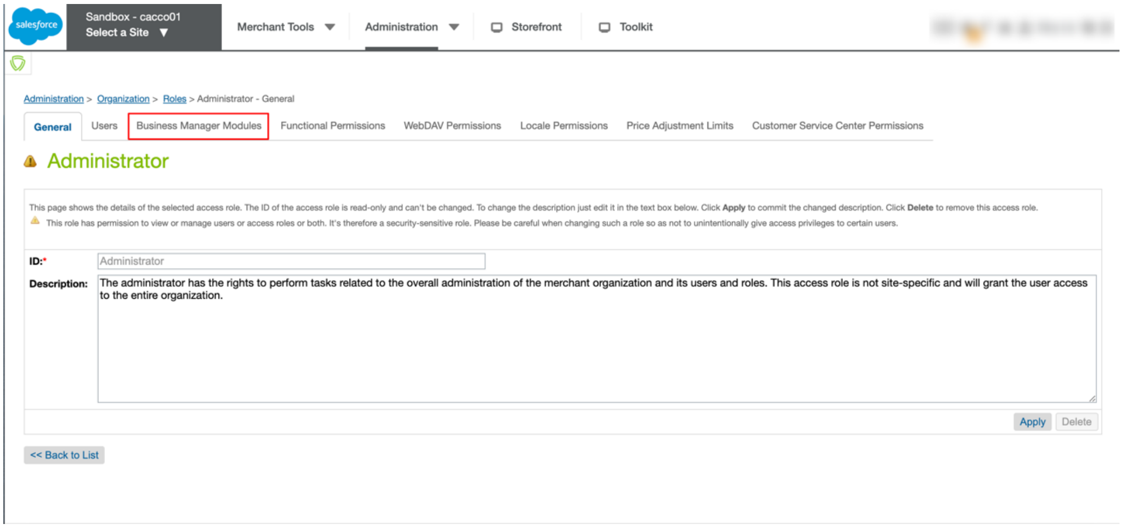
In order to use the Fraud Detection settings screen on the Bisiness Manager, go to **Administration > Organization > Role & Permissions**.



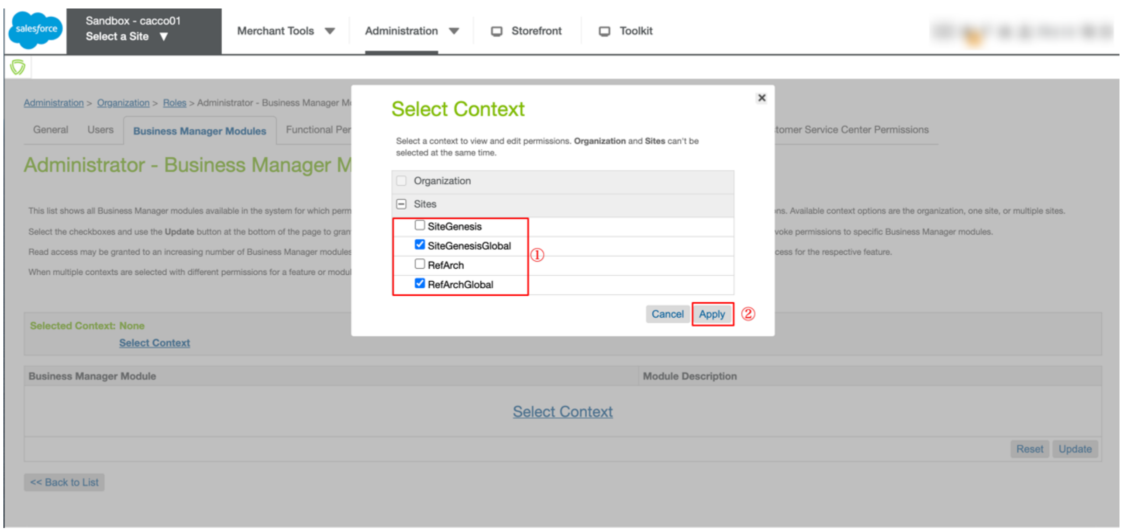
Select the ID to set (in this case , Administrator) .



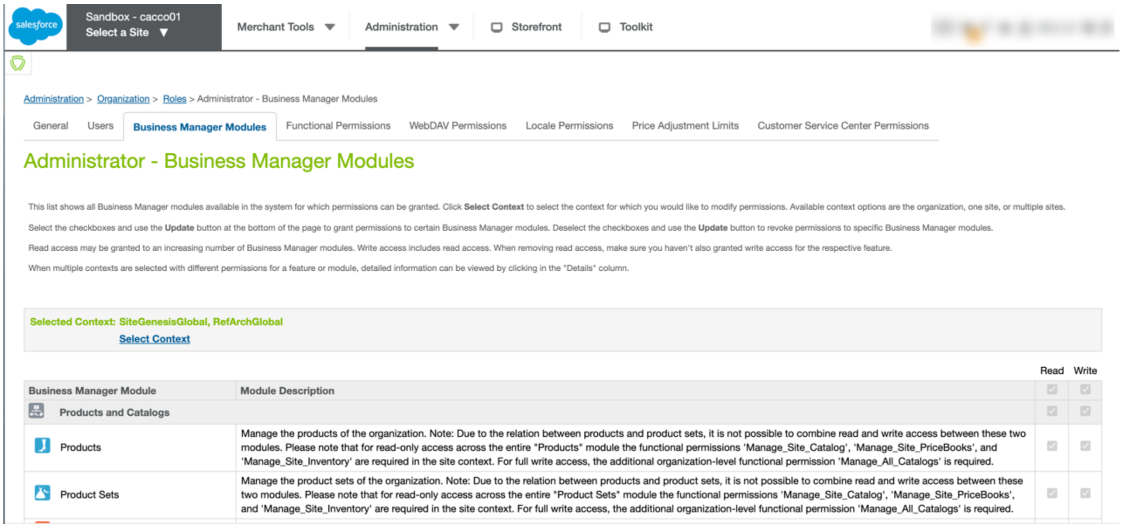
Select “**Business Manager Modules**” tab.



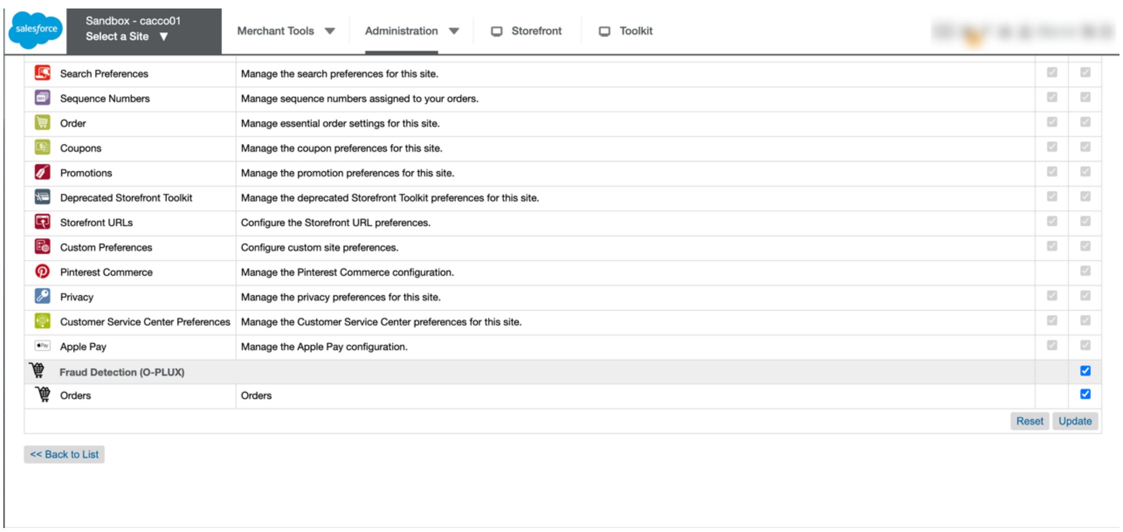
Select the target site.



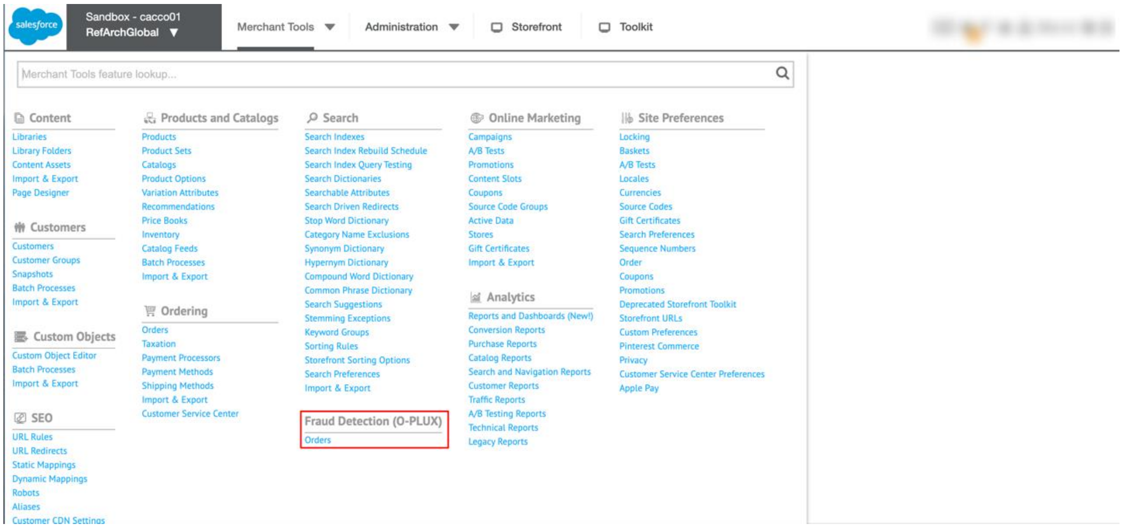
After selection, the following will be displayed



Check “Fraud Detection (O-PLUX)” settings options then “Update”.



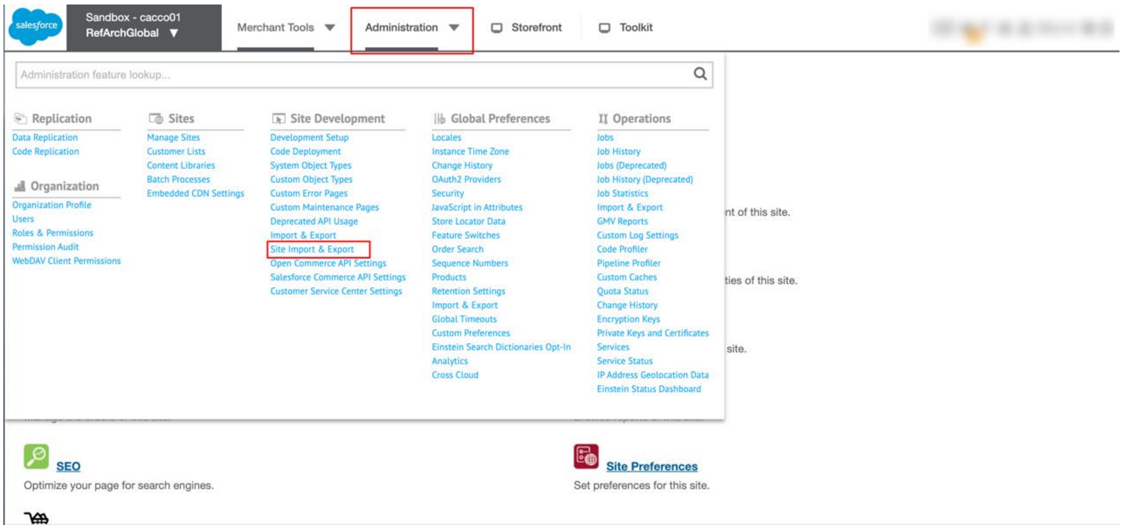
Now, “**Fraud Detection (O-PLUX)**” settings item will be shown under Merchant Tools menu.



3.2.3 Site Import

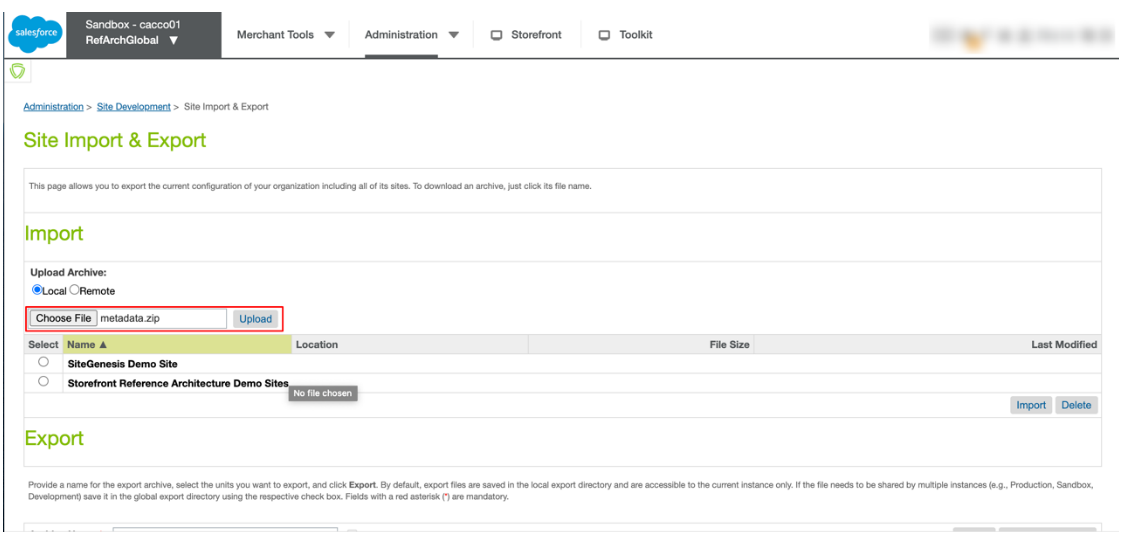
In Business Manager, go to

Administration > Site Development > Site Import & Export.

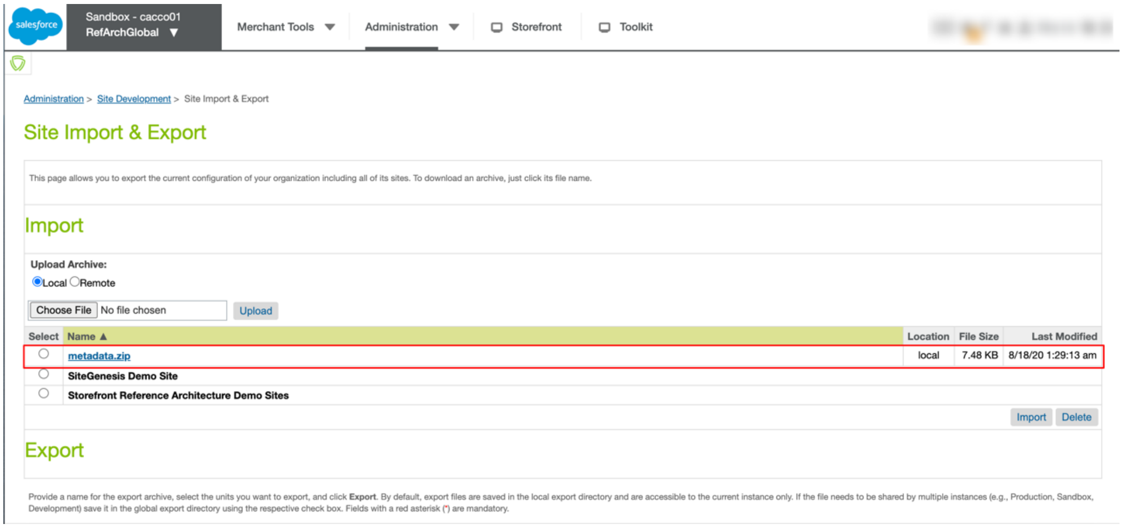


On the opened screen, select the file to upload for import. Select the zip file produced in step 3.1 of the guide and click the "Upload" button.

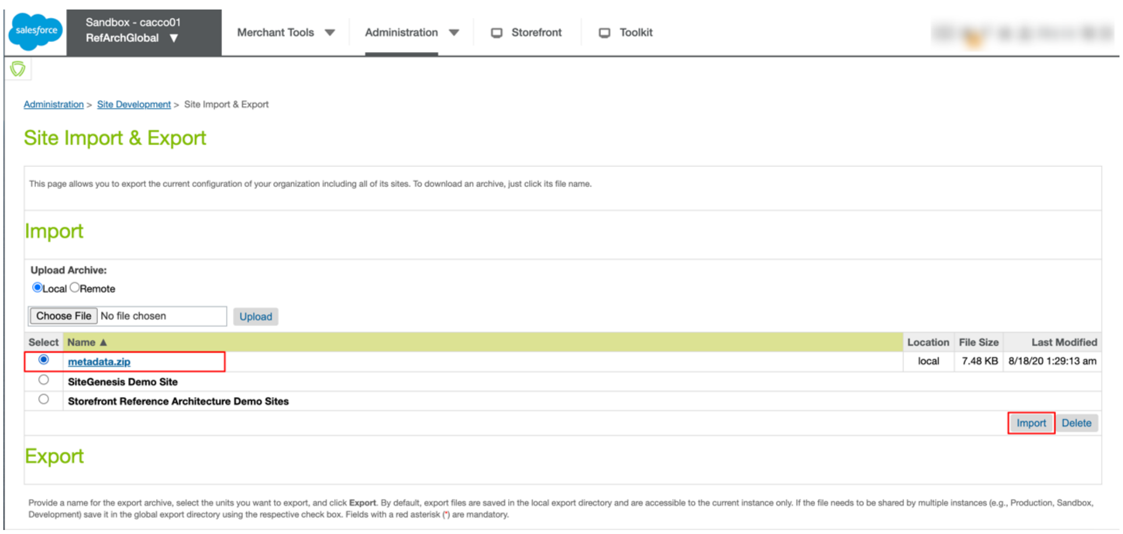
Setting target file: **{name of compressed file}.zip** and upload. In this case the target file is called "metadata.zip".



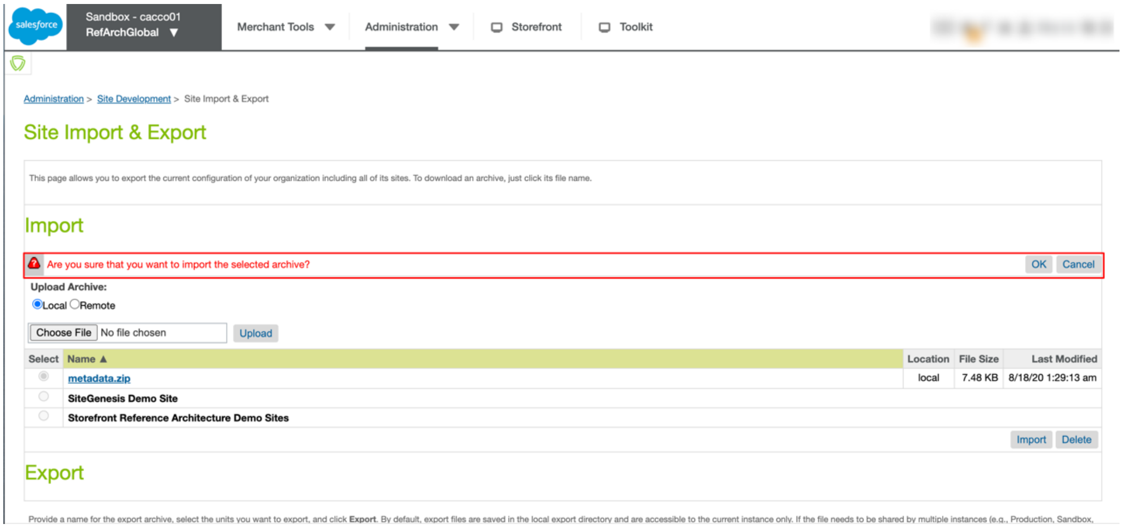
After the file is uploaded, the zip file will appear like this.



Select the uploaded file then click the “Import” button.



When the confirmation message appears, click the “OK” button. Wait a little bit for the file to be imported.



3.2.4 Metadata & Site Data Checking

After site import is completed, we should double check whether necessary items have been imported successfully.

- **Administration > Site Development > System Object Types > SitePreference > Attribute Definitions**

  oplux_shop_id

  oplux_appliction_id

  oplux_secret_access_id

  oplux_get_device_info_url

  oplux_run_background_job

  oplux_order_status_for_review_hold

  oplux_order_status_for_api_error_limit

- **Administration > Site Development > System Object Types > Order > Attribute Definitions**

  oplux_event_id

  oplux_settle_status

  oplux_rule_code

  oplux_rule_group

  oplux_rule_description

  oplux_event_aa_result

  oplux_response_result

  oplux_event_ma_result

  oplux_event_ma_result_memo

  oplux_blacked_flag

  oplux_black_customer_type

  oplux_black_other_category

oplux_black_settle_category

oplux_customer_notification_mail_sent_flag

oplux_error

oplux_device_info

oplux_skip_update_settle_status

oplux_event_api_response_json

- **Administration > Site Development > System Object Types > Basket > Attribute Definitions**

oplux_event_id

oplux_settle_status

oplux_rule_code

oplux_rule_group

oplux_rule_description

oplux_event_aa_result

oplux_response_result

oplux_event_ma_result

oplux_event_ma_result_memo

oplux_blacked_flag

oplux_black_customer_type

oplux_black_other_category

oplux_black_settle_category

oplux_customer_notification_mail_sent_flag

oplux_error

oplux_device_info

oplux_skip_update_settle_status

oplux_event_api_response_json

- **Administration > Site Development > System Object Types > PaymentMethod > Attribute Definitions**

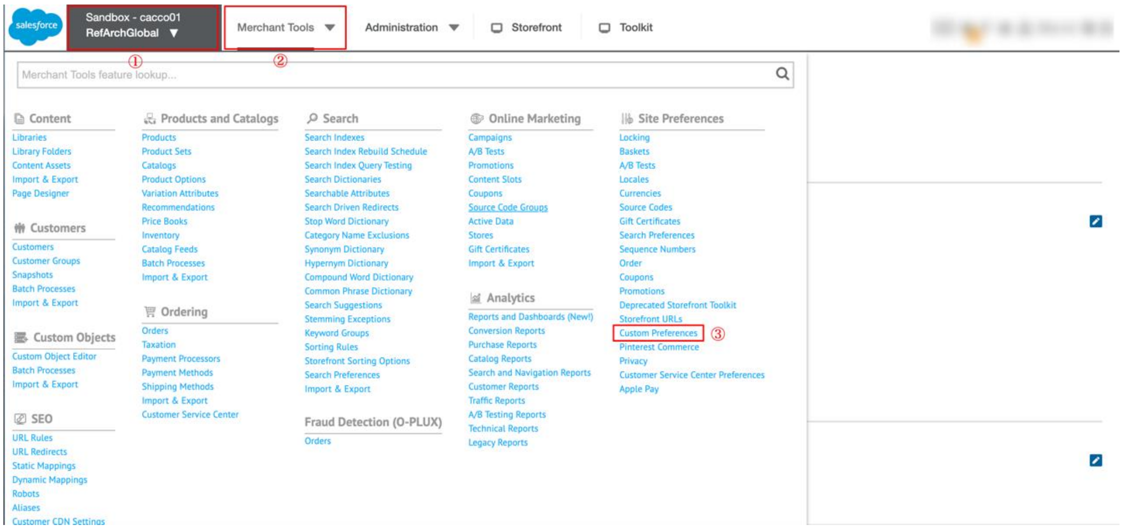
oplux_model_id

oplux_payment_method

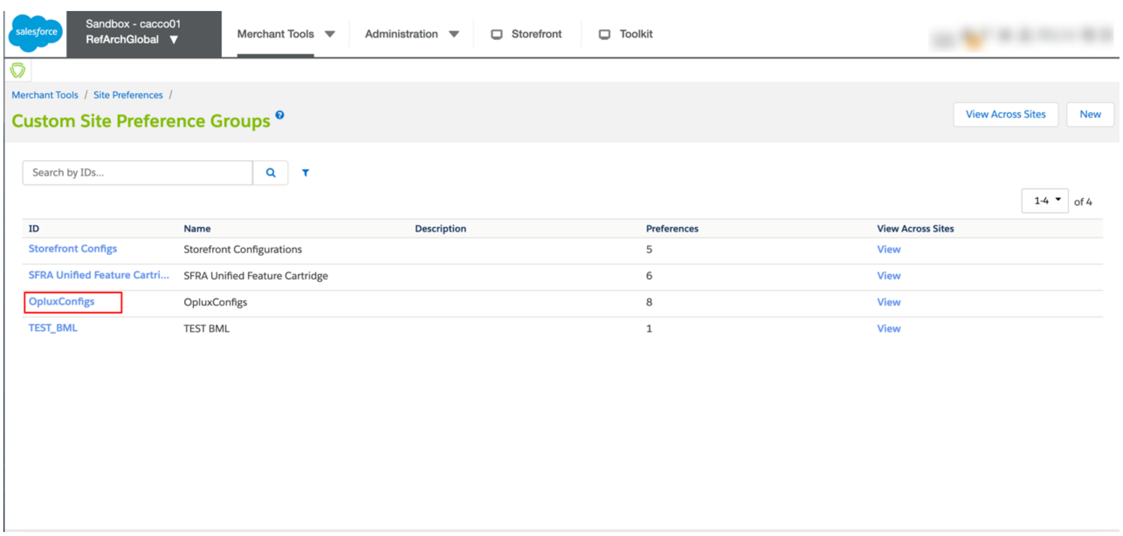
3.2.5 Custom Site Preference for Fraud Detection

Select the site in Business Manager, to use the fraud detection.

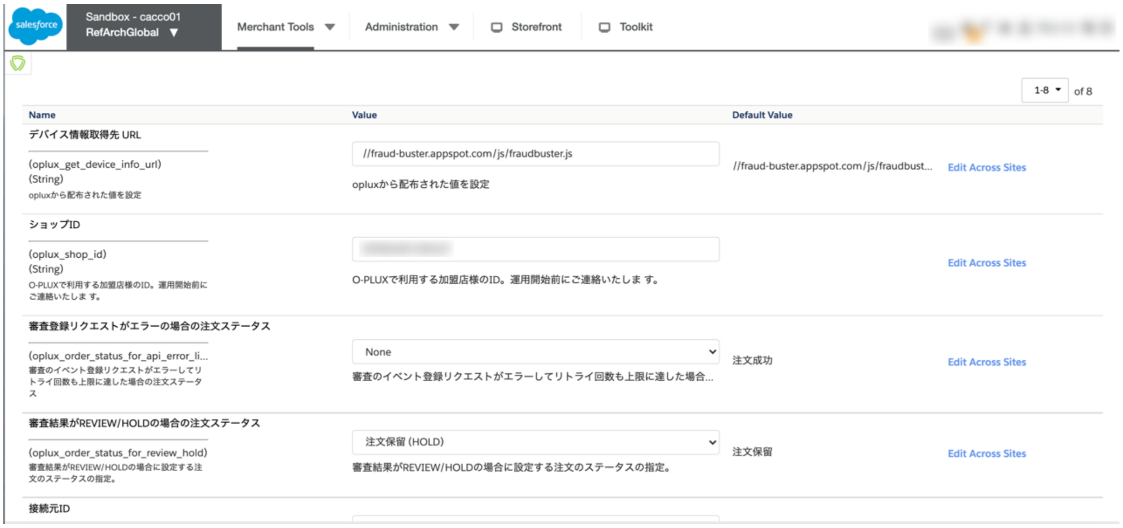
Go to **Merchant Tools > Site Preference > Custom Preference**.



Select the "**OpluxConfigs**".



Set the following items.



ショップ ID：**oplux_shop_id**

接続元 ID：**oplux_application_id**

シークレットキー: **oplux_secret_access_id**

デバイス情報取得 URL: **oplux_get_device_info_url**

Note: Each item is used for API authentication of the fraud detection.

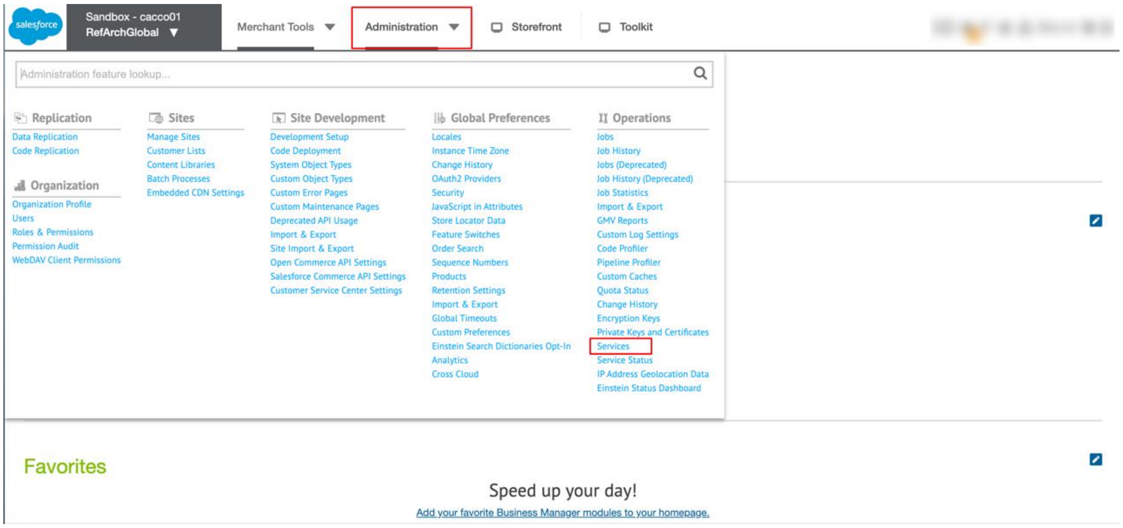
Please make sure to set the information provided by the service provider correctly.

3.2.5.1 OpluxConfigs items

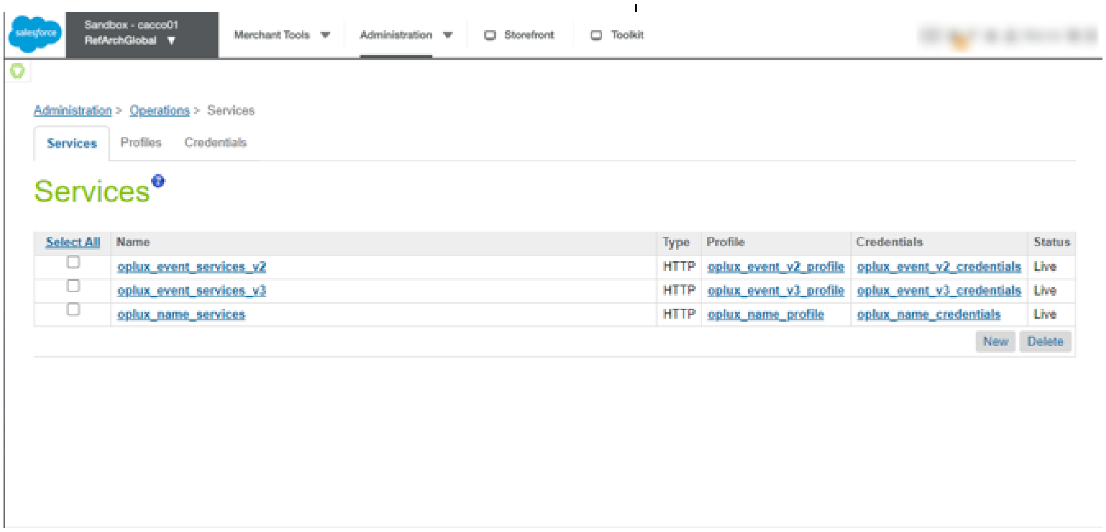
項目	説明
デバイス情報取得 URL	URL for getting end user's device info. Please set the value provided.
ショップ ID	ID used in O-PLUX system. Please set the value provided.
審査登録リクエストがエラーの場合の注文ステータス	Handling order status when event registration API error occurs.
自動審査結果が REVIEW/HOLD の場合の注文ステータス	Handling order status when event registration API result is REVIEW or HOLD.
接続元 ID	ID used for connection. Please set the value provided.
シークレットキー	Required for API authentication. Please set the value provided.
目視審査サービスを使用	Setting for active or deactive “Oplux_GetEventInfo” job. Check it if you want to active.

3.2.6 Services

In Business Manager, go to **Administration > Operations > Services**.



Check that the following items are displayed.



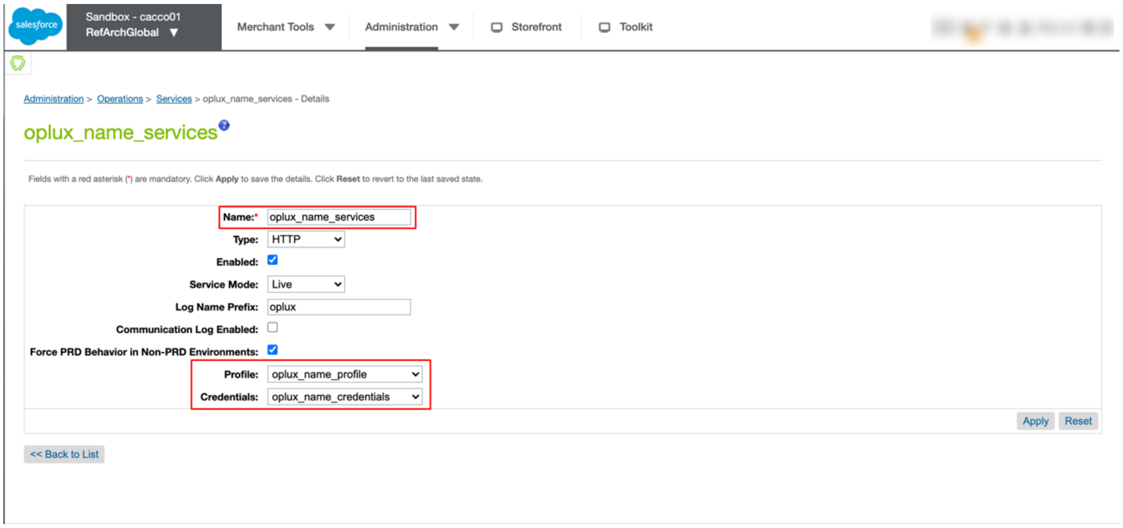
- name normalization API:

  name: **oplux_name_services**

  profile: **oplux_name_profile**

  Credentials: **oplux_name_credentials**

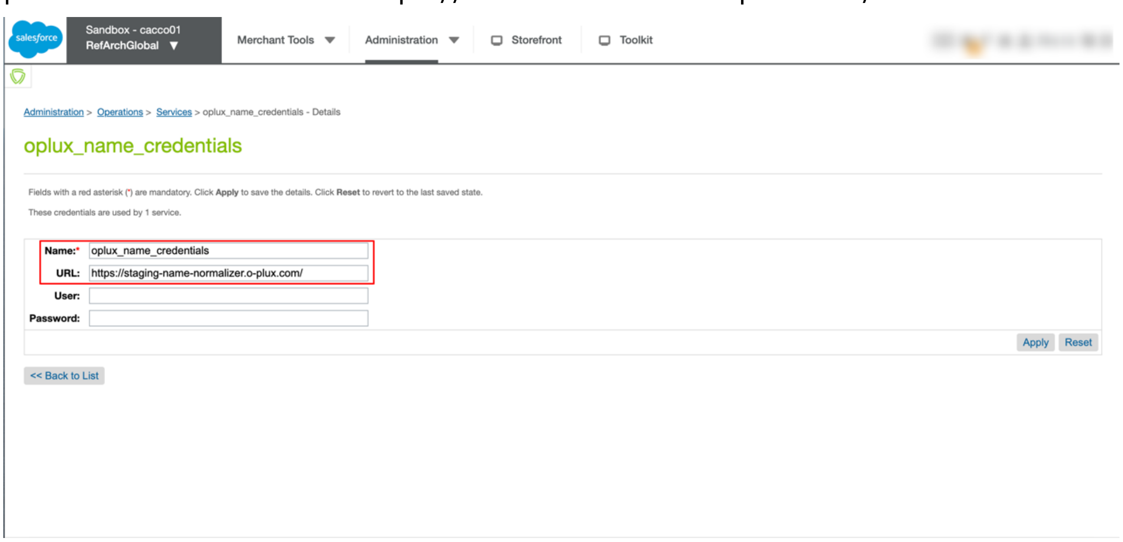
Click the link of "**oplux_name_services**", make sure the name, profile, and credentials fields are set.



Click the "**Back to List**" button, then click the link of "**oplux_name_credentials**". Make sure that value for name is set. For the URL item, set the information according to the environment. User and password are not required.

verification environment : https://staging-name-normalizer.o-plux.com/

production environment : https://name-normalizer.o-plux.com/



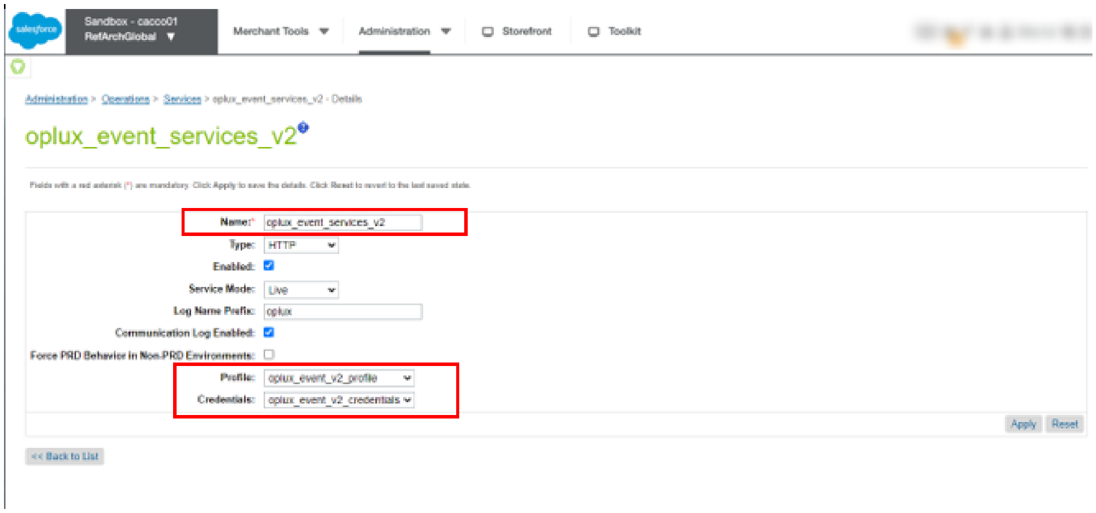
- Event acquisition / update（version2）：

  name: **oplux_event_services_v2**

  profile: **oplux_event_v2_profile**

  credentials: **oplux_event_v2_credentials**

Click the link of "**oplux_event_services_v2**", make sure the name, profile, and credentials fields are set.



Click the "**Back to List**" button, then click the link of "**oplux_event_v2_credentials**".

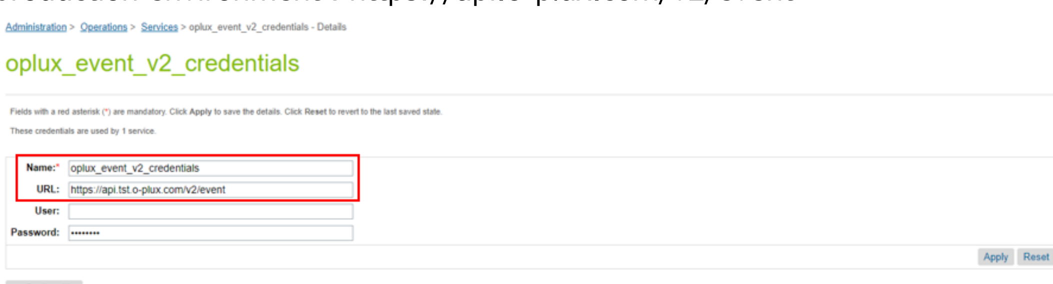
Make sure that value for name is set. For the URL item,

set the information according to the environment.

User and password are not required.

verification environment : https://api.tst.o-plux.com/v2/event

production environment : https://api.o-plux.com/v2/event



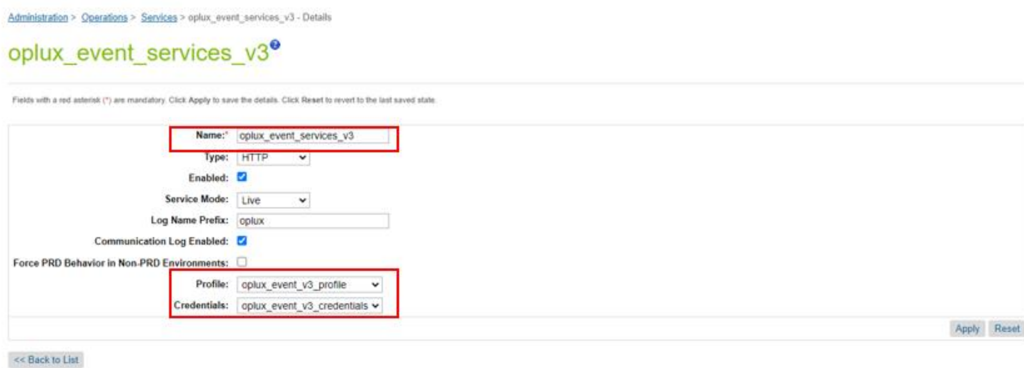
- Event registration（version3）：

    name: **oplux_event_services_v3**

    profile: **oplux_event_v3_profile**

    credentials: **oplux_event_v3_credentials**

Click the link of “**oplux_event_services_v3**”, make sure the name, profile, and credentials fields are set.



Click the “**Back to List**” button, then click the link of “**oplux_event_v3_credentials**”.

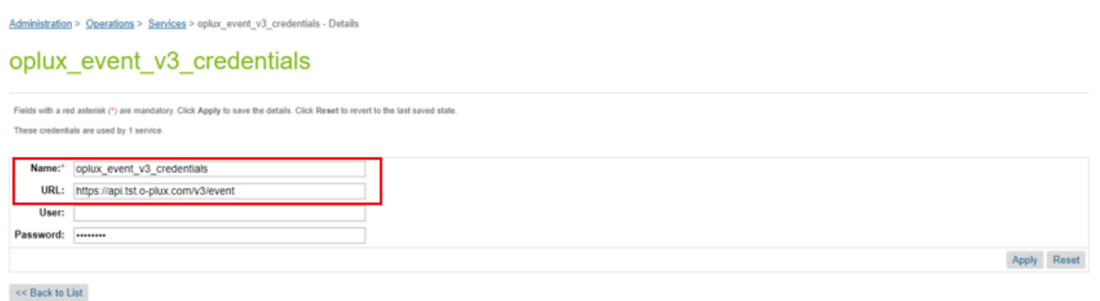
Make sure that value for name is set. For the URL item,

set the information according to the environment.

User and password are not required.

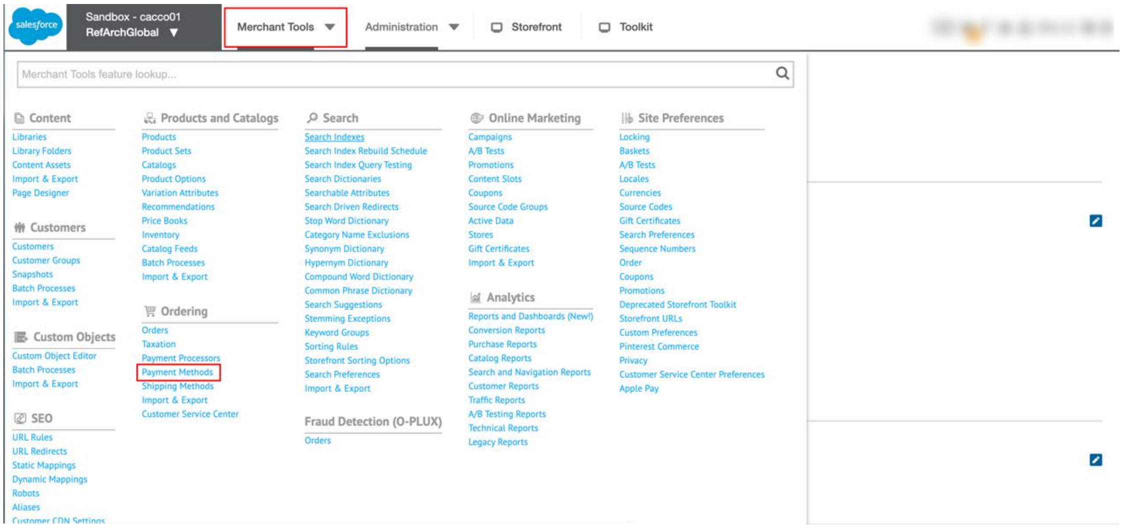
verification environment : https://api.tst.o-plux.com/v3/event

production environment : https://api.o-plux.com/v3/event

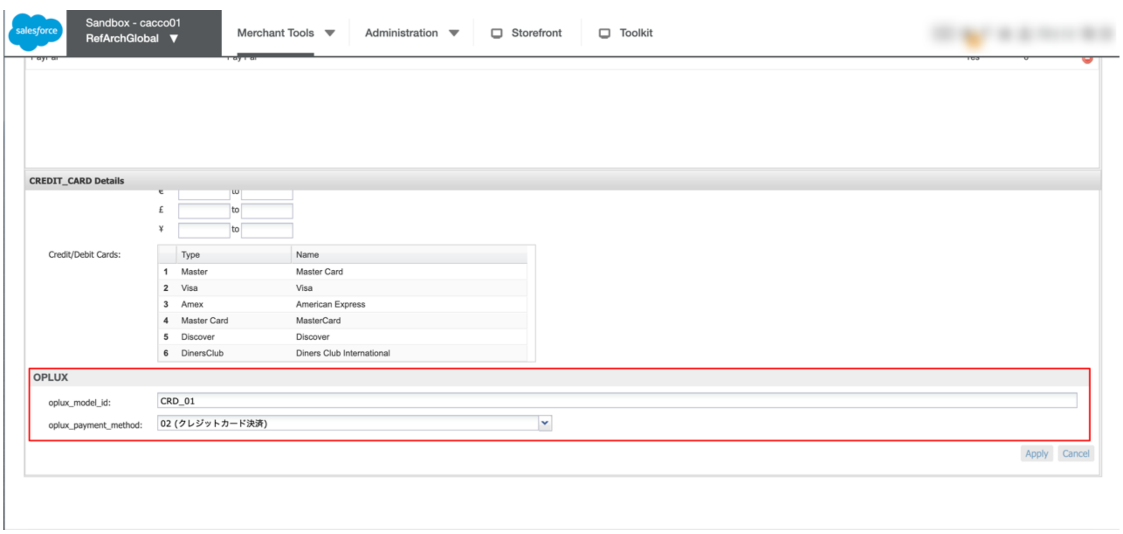


3.2.7 Payment Methods

In Business Manager, go to **Merchant Tools > Ordering > Payment Methods**



Check the "**OPLUX**" item for each payment methods must be set.



- **oplux_model_id**

For example: “CRD_01”, “ATB_01”, etc. It depends on the value provided.

- **oplux_payment_method**

Number	Payment method
01	Deferred payment
02	Credit payment
03	Cash on delivery
04	Prepayment
05	Electronic money
06	Point payment
07	Account transfer
08	Installments
09	Payeasy
10	PayPal
99	Other

3.3 Code Setup Guide

The below infomation shows the code that needs to be modified from SiteGenesis .

If you cannot introduce the "int_oplux_sfra_changes" cartridge as is, please modify the code by referring to the following guide.The existing code has a gray font color.

- Customize **cartridge/controllers/CheckoutServices.js**

This is the variable declaration section.

```
' use strict ';
var server = require('server’ );
var eventHelpers = require('*/cartridge/scripts/utils/eventHelpers');
var constants = require('*/cartridge/scripts/utils/constants');
server.extend (module.superModule);
var Transaction = require('dw/system/Transaction');
var BasketMgr = require('dw/order/BasketMgr');
var currentBasket = BasketMgr.getCurrentBasket();
```

Save the device information and the first number of the credit card for O-PLUX event registration API and execute the event registration API.

If the examination result is OK, the examination result is cached for each examination model. If the examination result is NG, the cache is cleared and an error screen is displayed.

```
'SubmitPayment'
function (req, res, next) {
  // Do not review if validation error occurs
   if (res.viewData.error) {
```

```
          return next();
        }
        var deviceInfo = request.httpParameterMap.get('fraudbuster');
        var paymentForm = server.forms.getForm('billing');
        var cardNumber = paymentForm.creditCardFields.cardNumber.value;
       var opluxCallHelpers = require('*/cartridge/scripts/utils/opluxCallHelpers')
;
      var URLUtils = require('dw/web/URLUtils')
(deviceInfo){           deviceInfo.getValue().length <=
constants.MAX_SESSION_STRING_LENGTH ? req.session.privacyCache.set(
, deviceInfo.getValue()) : req.session.privacyCache.set(opluxDeviceInfo, '');
            req.session.privacyCache.set('cardNumber', cardNumber);
        }
      ...
        return next();
    }
);
```

Before creating an Order, update the results of the event Registration API for the current judging model in Basket.This will preserve the results of the event registration API in the Order.

```
server.prepend(
   'PlaceOrder',
   function (req, res, next) {
      var BasketMgr = require(' dw/order/BasketMgr');
      var currentBasket = BasketMgr.getCurrentBasket();
      eventHelpers.updateBasketOpluxExaminationResult(currentBasket);
      return next();
   }
```

After creating the Order, verify that the event registration API results are correctly updated to the Order.

```
server.append (
    'PlaceOrder',
    server.middleware.post,
    server.middleware.https,
    function (req, res, next) {
        ...
        var OpluxCallHelpers =
require('/int_oplux/cartridge/scripts/utils/opluxCallHelpers');
        ...
        ...
        if(orderID){
          var order =OrderMgr.getOrder(orderID);
          if (order && viewData.error) {
             var isValid = OpluxCallHelpers.opluxResultValidation(order);
             if (!isValid) {
                 res.json({
                     error: true,
                     cartError: true,
                     redirectUrl: URLUtils.url('Page-Show', 'cid', "opluxEventNG").toString()
                 });
              }
          }
          // OPLUX: process order status by Oplux
          OpluxCallHelpers.postProcess(order);
          // clear orderNo from cache
          session.privacy.orderNo = null;
          session.privacy.model_id = null;
       }
        return next();
```

```
  }
);
```

- Customize **cartridge/template/default/checkout/billing/billing.isml**

Set the URL to get the device information.

```
...
<!-------------------------------------------------------------------------->
<!-- oplux get device info                                              -->
<!-------------------------------------------------------------------------->
<fieldset>
  <input type="hidden" id="fraudbuster" name="fraudbuster"/>
  <script type="text/javascript" src="
${dw.system.Site.current.preferences.custom.oplux_get_device_info_url}"></script>
</fieldset>
...
```

3.4 Linking Settlement Status

If you want to add “deferred-payment” etc. to the payment method and link the settlement status, please add or modify the code in the following resource file. If you want to use the “overdue” setting in O-PLUX, you need to set the payment deadline.

Please refer to the sample code below for customization.

- Customize **cartridge/scripts/jobs/opluxUpdateSettleStatus.js**

Get order information to update the status of O-PLUX.

```
function searchPastDueForPaymentOrders() {
      var OrderMgr = require('dw/order/OrderMgr');
      var Constants = require('*/cartridge/scripts/utils/constants');

      var orders = OrderMgr.searchOrders(
            "custom.oplux_event_id != NULL AND custom.oplux_skip_update_settle_status != TRUE
AND " +
            "custom.oplux_settle_status = {0}",
            'creationDate asc',
            Constants.EVENT_REQUEST_SETTLE_DEFAULT_STATUS.PAYMENT_OVERDUE
       );
      return orders;
}
```

- Customize **cartridge/scripts/jobs/opluxSettleStatusFinder.js**

Returns the settlement status linked to the O-PLUX.

Define a function for each audit model ID in the CustomStatusMap property. The sample code for the case of "oplux_model_id: 01" is implemented in the cartridge. For example, if you want to add "oplux_model_id: 02" as a payment method, please refer to the commented out part below and add processing according to the payment method.

```
customStatusMap: {
  '01': function (order) {
```

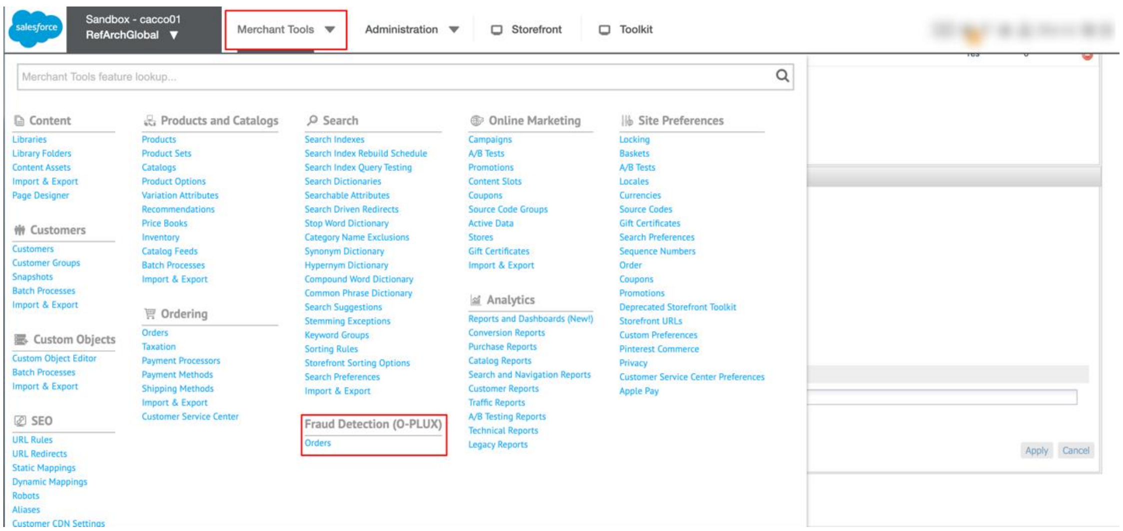
```
    var Order = require('dw/order/Order');
    Logger.debug("=====Custom SettleStatusFinder Started.=====");
    var now = new Date();
    if(order.custom.payLimitDate < now &&
            order.paymentStatus == Order.PAYMENT_STATUS_NOTPAID &&
            (order.status == Order.ORDER_STATUS_NEW ||
            order.status == Order.ORDER_STATUS_OPEN)
     ){
       return '30';
    }
     return null;
  },
  // If 02 is added, add the following code
  '02': function (order) {
    // Describe the process according to the payment method
  }
  // If 02 is added, add the above code
},
```

3.5 How to Use Fraud Detection (O-PLUX)

3.5.1 Order List

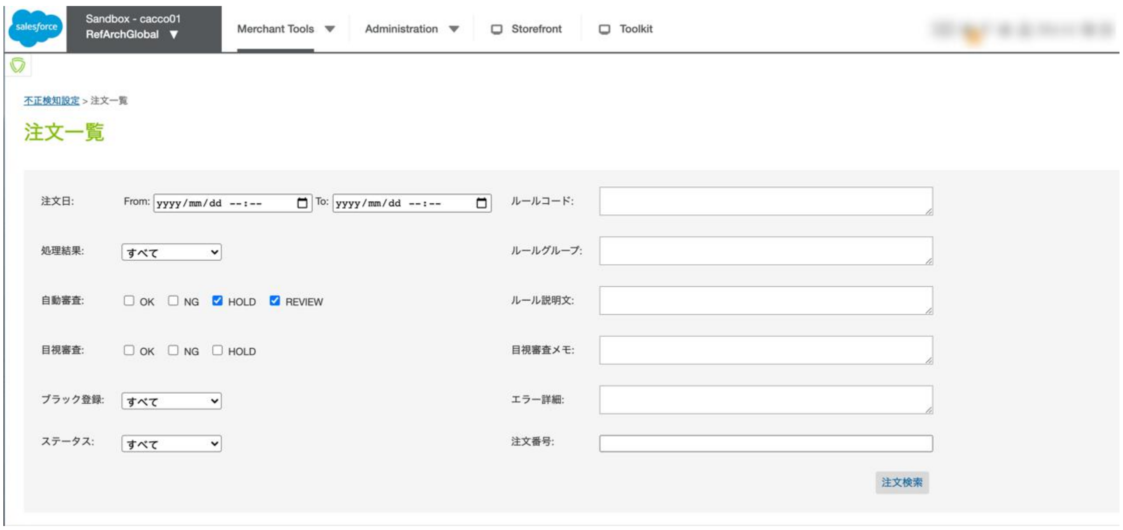
Select the site in Business Manager, to use the fraud detection.

Go to **Merchant Tools > Fraud Detection (O-PLUX) > Orders** to see the list of fraud detected orders.



3.5.2 Search

You can search for fraudulent orders by setting conditions such as order date and processing result. By default, the conditions for automatic examination items are "**HOLD**" or "**REVIEW**"



Order numbers are displayed starting with the most recent order.

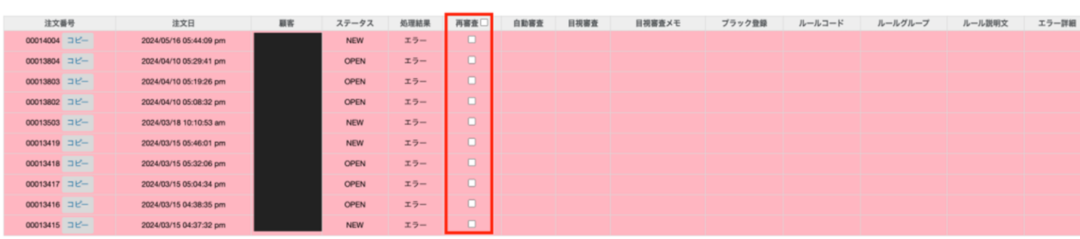
3.5.3 Update

3.5.3.1 Re-examination

If the processing result is “Error”, you can re-examine.Orders with order status "CANCELLED", "FAILED" or "COMPLETED" cannot be re-examined.
You can also re-examine if the export status is “EXPORTED”.

Set the processing result item to “Error” and click the “Order Search” button.

Check box is displayed for orders that can be re-examined.



You can re-examine by selecting the check box and click the “Re-examination” button.

注文番号	注文日	顧客	ステータス	処理結果	再審査 ☐	自動審査	目視審査	目視審査メモ	ブラック登録	ルールコード	ルールグループ	ルール説明文	エラー詳細
00014004 コピー	2024/05/16 05:44:09 pm	[redacted]	NEW	エラー	☑								
00013804 コピー	2024/04/10 05:29:41 pm		OPEN	エラー	☑								
00013803 コピー	2024/04/10 05:19:26 pm		OPEN	エラー	☑								
00013802 コピー	2024/04/10 05:08:32 pm		OPEN	エラー	☑								
00013503 コピー	2024/03/18 10:10:53 am		NEW	エラー	☑								
00013419 コピー	2024/03/15 05:46:01 pm		NEW	エラー	☐								
00013418 コピー	2024/03/15 05:32:06 pm		OPEN	エラー	☐								
00013417 コピー	2024/03/15 05:04:34 pm		OPEN	エラー	☐								
00013416 コピー	2024/03/15 04:38:35 pm		OPEN	エラー	☐								
00013415 コピー	2024/03/15 04:37:32 pm		NEW	エラー	☐								

再審査

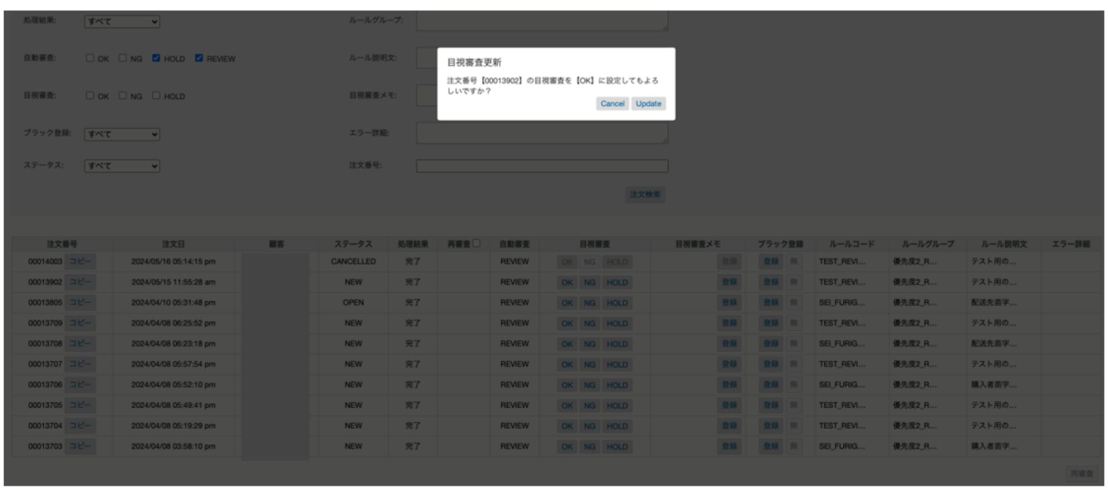
3.5.3.2 Manual examination

You can set the result of Manual examination by selecting the “OK”, “NG”, “HOLD” button.

Conditions:

- Order status “NEW” “OPEN” “REPLACED”
- Export status other than “EXPORTED”

Select one of the buttons, then confirmation popup will appear. It can be set by clicking the “Update” button.



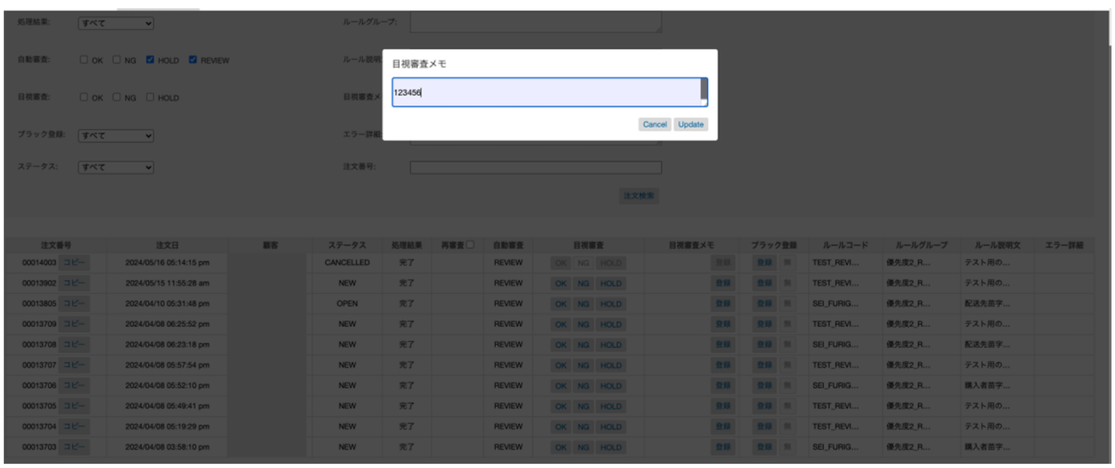
3.5.3.3 Manual examination Memo

You can register a Manual examination memo by clicking the “Register” button.

Conditions:

- Order status “NEW” “OPEN” “REPLACED”
- Export status other than “EXPORTED”

Click the “Register” button, the input field for the Manual examination memo will be displayed. After input the contents, it can be set by clicking the “Update” button.



3.5.3.4 Blacklist Registration

You can set the blacklist by clicking the “Register” button in the blacklist registration item.

By clicking the “Register” button, a popup to set reasons for blacklist registration will be displayed. Click the “Update” button to register.

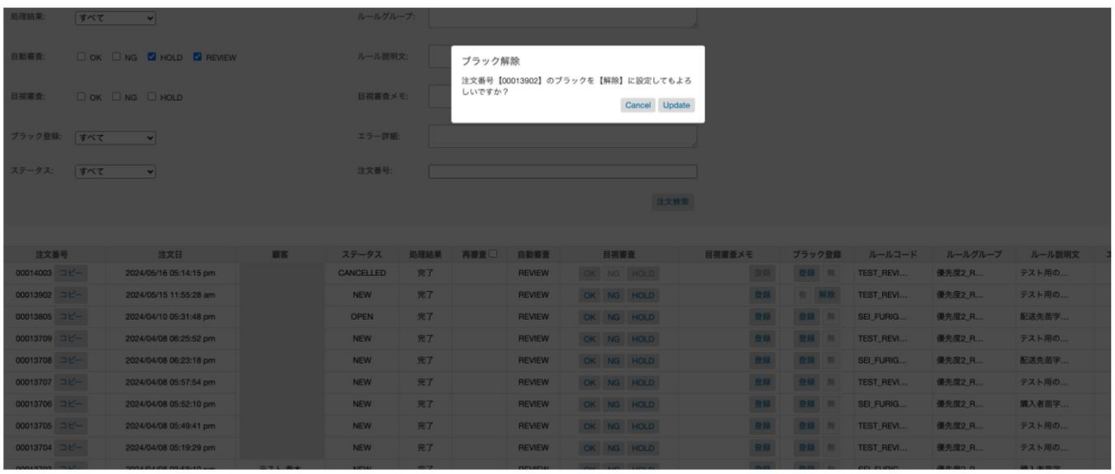
3.5.3.5 Blacklist Release

When one item is registered in blacklist, the “Release” button will be displayed.

You can release the registered item by clicking this button.

Click the “Release” button to display confirmation popup.

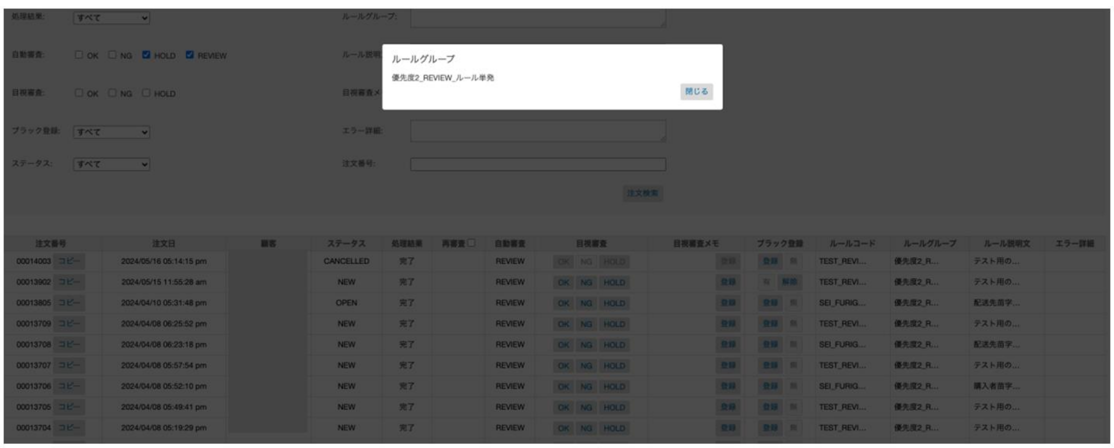
This item will be released by clicking the “Update” button.



3.5.3.6 Other

Click the items of “rule code”, “rule group”, “rule description”, and “error details”, then contents of each item are displayed.

If you click a “rule group” item, it will be displayed as follows:



3.6 Jobs

3.6.1 Manual examination result acquisition - **Oplux_GetEventInfo**

When using the Manual examination service, the result of the Manual examination of the target order is acquired from O-PLUX and linked to Business Manager.

The default setting is to run the job every 5 minutes.

- Target order

  Order information within 48 hours.

  Automatic examination item is “HOLD” or “REVIEW”.

  The result of Manual examination is not set.

Example :

注文番号	注文日	顧客	ステータス	処理結果	再審査	自動審査	目視審査	目視審査メモ	ブラック登録	ルールコード	ルールグループ	ルール説明文	エラー詳細
00014005 コピー	2024/05/16 05:56:16 pm	二郎 テスト	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	
00014003 コピー	2024/05/16 05:14:15 pm	二郎 テスト	CANCELLED	完了		REVIEW	OK NG HOLD	登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	
00013902 コピー	2024/05/15 11:55:28 am	太郎 テスト	NEW	完了		REVIEW	OK NG HOLD	登録	有 解除	TEST_REVI...	優先度2_R...	テスト用の...	
00013805 コピー	2024/04/10 05:31:48 pm	TEST AOKI	OPEN	完了		REVIEW	OK NG HOLD	登録	登録 無	SEI_FURIG...	優先度2_R...	配送先苗字...	
00013709 コピー	2024/04/08 06:25:52 pm	TEST AOKI	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	
00013708 コピー	2024/04/08 06:23:18 pm	TEST AOKI	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	SEI_FURIG...	優先度2_R...	配送先苗字...	
00013707 コピー	2024/04/08 05:57:54 pm	TEST AOKI	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	

After the job is executed, the contents of “Manual examination” and “Manual examination memo” of O-PLUX are linked to SFCC.

注文番号	注文日	顧客	ステータス	処理結果	再審査	自動審査	目視審査	目視審査メモ	ブラック登録	ルールコード	ルールグループ	ルール説明文	エラー詳細
00014005 コピー	2024/05/16 05:56:16 pm	二郎 テスト	NEW	完了		REVIEW	OK NG HOLD	5分ジョブ... 登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	
00014003 コピー	2024/05/16 05:14:15 pm	二郎 テスト	CANCELLED	完了		REVIEW	OK NG HOLD	登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	
00013902 コピー	2024/05/15 11:55:28 am	太郎 テスト	NEW	完了		REVIEW	OK NG HOLD	登録	有 解除	TEST_REVI...	優先度2_R...	テスト用の...	
00013805 コピー	2024/04/10 05:31:48 pm	TEST AOKI	OPEN	完了		REVIEW	OK NG HOLD	登録	登録 無	SEI_FURIG...	優先度2_R...	配送先苗字...	
00013709 コピー	2024/04/08 06:25:52 pm	TEST AOKI	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	
00013708 コピー	2024/04/08 06:23:18 pm	TEST AOKI	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	SEI_FURIG...	優先度2_R...	配送先苗字...	
00013707 コピー	2024/04/08 05:57:54 pm	TEST AOKI	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	
00013706 コピー	2024/04/08 05:52:10 pm	しゃ こうにゅう	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	SEI_FURIG...	優先度2_R...	購入者苗字...	
00013705 コピー	2024/04/08 05:49:41 pm	テスト テスト	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	
00013704 コピー	2024/04/08 05:19:29 pm	テスト テスト	NEW	完了		REVIEW	OK NG HOLD	登録	登録 無	TEST_REVI...	優先度2_R...	テスト用の...	

3.6.2 Settlement Status Sending - **Oplux_UpdateSettleStatus**

The settlement status of SFCC will be linked to O-PLUX. The default setting is to run the job every minute.

For the "irregular sample" and "customization (when deferred payment is selected)," sample code exists (see Section 3.4, " Linking Settlement Status "), and the merchant must customize the code based on the sample code. In addition, orders that have selected credit card payment and the payment status has been changed to "Paid" are not eligible for jobs. When credit card payment is selected, the timing for debiting the billing amount varies among credit card companies, making it difficult for the merchant to know when the amount will be debited.Therefore, orders with the above conditions will not be renewed on the job.

Settlement status pattern.

- Standard specifications

Pattern	Settlement status
Credit order	00:Before billing
Non-credit orders（Not paid)	00:Before billing
Non-credit orders（paid)	00:Before billing → 20:Paid
Manual examination “NG”	00:Before billing
Not shipped cancel（Not paid)	00:Before billing → 99:Cancelled
Shipped cancel（Not paid)	00:Before billing → 100:Cancelled after shipped
Not shipped cancel（Non-credit ,Paid）※1	00: Before billing → 20:Paid →99:Cancelled
Not shipped cancel （Credit ,Paid）※2	00: Before billing →99:Cancelled
Shipped cancel（Non-credit ,Paid）※1	00:Before billing → 20:Paid →100:Cancelled after shipped

Shipped cancel（Credit ,Paid）※2	00: Before billing →100:Cancelled after shipped
Order failed（process result “10:success”, code error）	00:Before billing → 99:Cancelled
Order failed（API process result 20:error, no response）	Not linked because the event ID has not been acquired
Order failed（Automatic examination HOLD/REVIEW）	00:Before billing → 99:Cancelled

※1 : When you select a payment method other than credit card, such as deferred payment, the payment status will be updated to "00:Before Billing → 20:Paid ⇨ 99: Cancelled or 100: Cancelled after shipped ".

※2 : When you select a credit card as your payment method, the payment status will be updated to "00: Before billing →99:Cancelled or 100: Cancelled after shipped ".

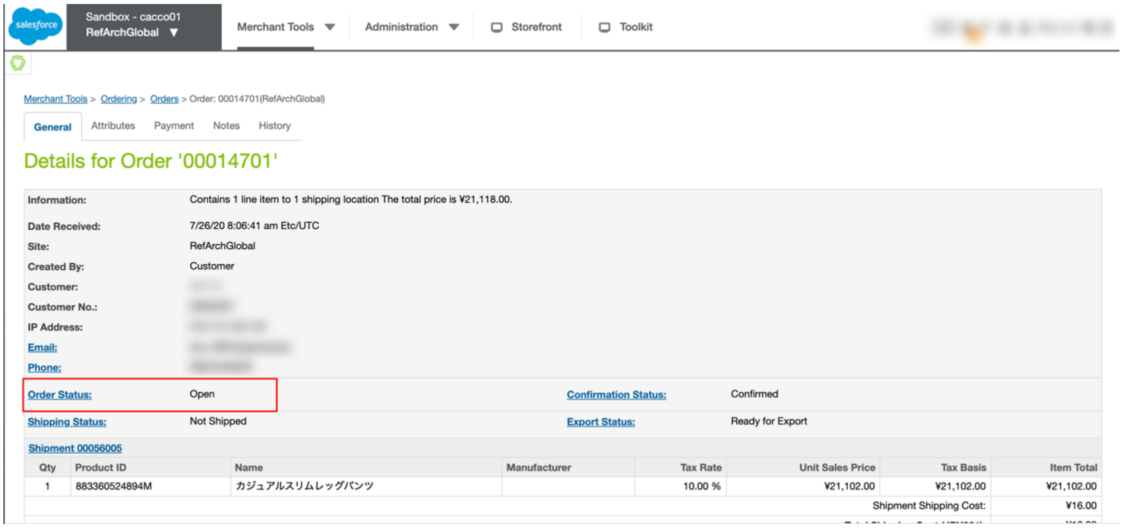
- Irregular sample

Pattern	Settlement status
Automatic examination “OK” → Non-credit,Paid → Manual examination “NG”	00:Before billing → 20:Paid

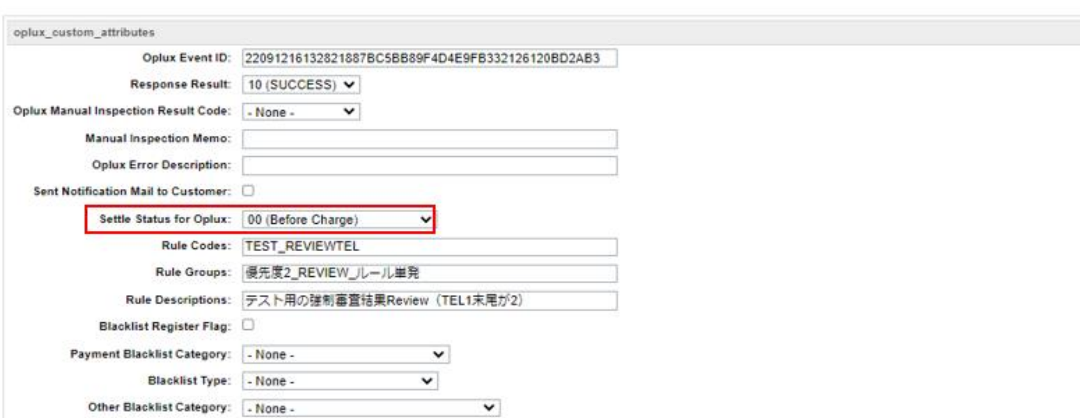
- Customization (Deferred payment)

Pattern	Settlement status
Expired payment	00:Before billing → 30:Overdue
Payment overdue	00:Before billing → 30:Overdue → 20:Paid
Cancellation du to payment expiration	00:Before billing → 30:Overdue → 100:Cancelled after shipped

Example: pre-payment cancellation, it is as follows.

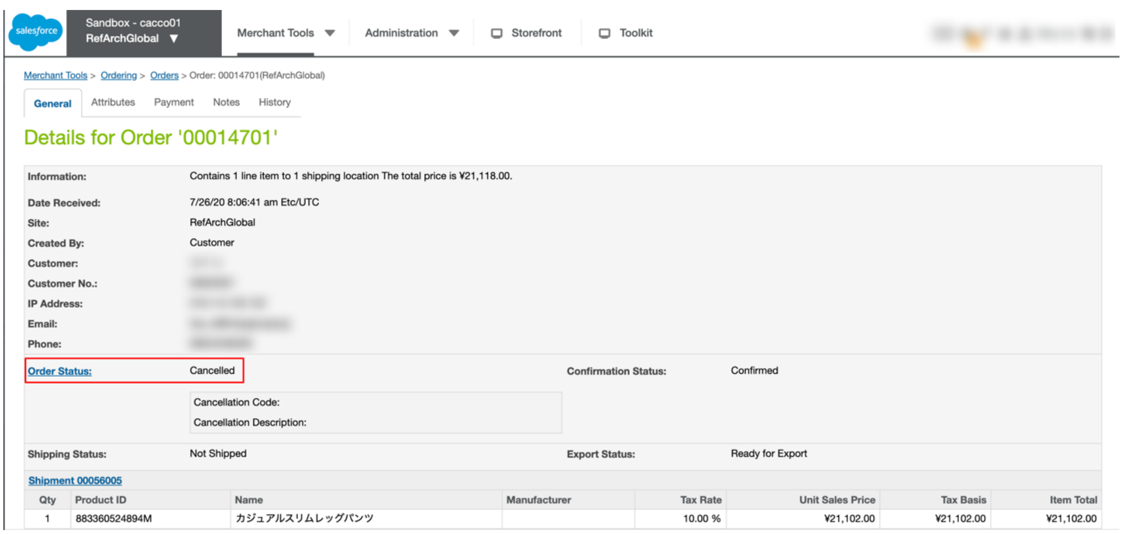


The final linked settlement status on the "**Attribute**" tab is "00 (before billing)".

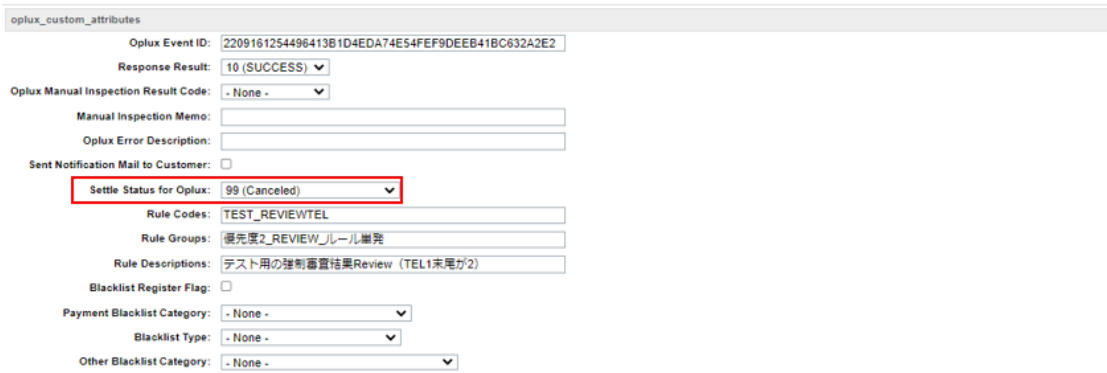


The settlement status on the O-PLUX management screen is “before billing”.

SFCC order status is changed to “Cancelled”.



The final linked settlement status on the “**Attribute**” tab will be changed “99 (Cancel)”.



The settlement status on the O-PLUX management screen will be changed to “Cancel”.

3.7 Customize parameters

Some parameters of O-PLUX are not supported in the SFCC default settings.
If you want to use these parameters, you need to add custom attributes and modify the source code.

3.7.1 O-PLUX parameters not supported by SFCC

The following O-PLUX parameters are not targeted for transmission in API requests for the O-PLUX cartridge.Check with your service provider for details and interface specifications.

- Language : **info.lang**
- Time zone : **info.tz**
- Currency : **info.currency**
- Fields : **telegram.fields**
- Last 4 digits of card number :
  telegram.event.ec.settle.credit_card.number_hash
- Card authorization information :
  telegram.event.ec.settle.credit_card.authori_info
- Authorization number :
  telegram.event.ec.settle.credit_card.authori_info.authori_code
- Credit card issuer :
  telegram.event.ec.settle.credit_card.issuer_foreign_flg
- Buyer first name hash :
  telegram.event.ec.customers.buyer.hashed_name.first_name

- Buyer normalized first name hash : **telegram.event.ec.customers.buyer.hashed_name.normalized_first_name**
- Buyer last name hash : **telegram.event.ec.customers.buyer.hashed_name.last_name**
- Buyer normalized last name hash : **telegram.event.ec.customers.buyer.hashed_name.normalized_last_name**
- Buyer city : **telegram.event.ec.customers.buyer.address.addressB**
- Buyer email account hash１ : **telegram.event.ec.customers.buyer.email.pc.hashed_account**
- Buyer email information２ : **telegram.event.ec.customers.buyer.email.mobile**
- Buyer email account hash２ : **telegram.event.ec.customers.buyer.email.mobile.hashed_account**
- Buyer email account sha2 hash２ : **telegram.event.ec.customers.buyer.email.mobile.hashed_account_sha2**
- Buyer email domain２ : **telegram.event.ec.customers.buyer.email.mobile.domain**
- Buyer department : **telegram.event.ec.customers.buyer.company.department**
- Customer category : **telegram.event.ec.customers.buyer.existing_customer_flg**
- Specified deliver datetime : **telegram.event.ec.customers.deliveries.specified_deliver_datetime**

- Delivery specified existence : **telegram.event.ec.customers.deliveries.delivery_specified_existence**
- Delivery service : **telegram.event.ec.customers.deliveries.delivery_company_name**
- Delivery first name hash : **telegram.event.ec.customers.deliveries.hashed_name.first_name**
- Delivery normalized first name hash : **telegram.event.ec.customers.deliveries.hashed_name.normalized_first_name**
- Delivery last name hash : **telegram.event.ec.customers.deliveries.hashed_name.last_name**
- Delivery normalized last name hash : **telegram.event.ec.customers.deliveries.hashed_name.normalized_last_name**
- Delivery city : **telegram.event.ec.customers.deliveries.address.addressB**
- Delivery email account hash 1 : **telegram.event.ec.customers.deliveries.email.pc.hashed_account**
- Delivery email information 2 : **telegram.event.ec.customers.deliveries.email.mobile**
- Delivery email account hash 2 : **telegram.event.ec.customers.deliveries.email.mobile.hashed_account**
- Delivery email account sha2 hash 2 : **telegram.event.ec.customers.deliveries.email.mobile.hashed_account_sha2**
- Delivery email domain 2 : **telegram.event.ec.customers.deliveries.email.mobile.domain**

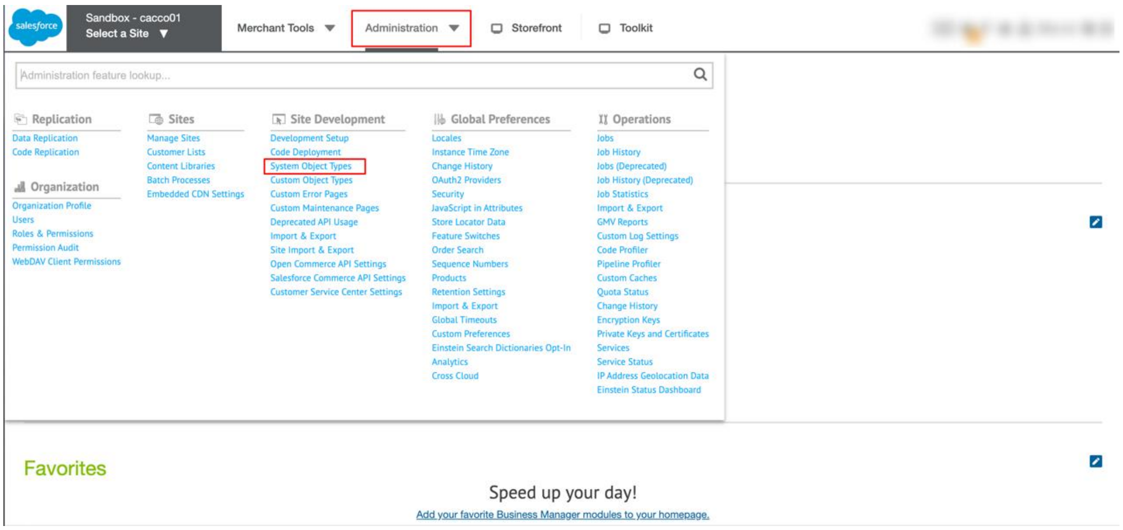
- Delivery department :
  telegram.event.ec.customers.deliveries.company.department
- Stock availability : **telegram.event.ec.items.item_stock_flg**
- Shop tenant Information : **telegram.event.ec.tenant**
- Shop tenant ID : **telegram.event.ec.tenant.shop_tenant_id**
- Shop tenant limit price : **telegram.event.ec.tenant.tenant_limit_price**
- Credit line change ratio :
  telegram.event.ec.tenant.member_limit_change_ratio
- Shop tenant name : **telegram.event.ec.tenant.tenant_name**
- Shop tenant URL : **telegram.event.ec.tenant.tenant_url**
- Shop tenant registration date :
  telegram.event.ec.tenant.tenant_registration_date
- Shop tenant website ID : **telegram.event.ec.tenant.site_id**
- Shop tenant website password : **telegram.event.ec.tenant.site_pass**
- Last name romaji : **lastRomajiFurigana**

※Some items, such as the name of the purchaser's company, are set to be requested by the O-PLUX API, but are not SFRA standard items that can be entered in Storefront, so merchants will need to modify them if necessary.
※In accordance with SFCC security, API requests for "O-PLUX" are configured as Hash TypeB requests using SHA256. No API items related to Hash TypeA are set.
※For more information, contact Cacco Inc.
※The custom attribute ID for "last name furigana" may be different for each merchants, but the cartridge is implemented on the assumption that the custom attribute ID "oplux_last_name_kana" exists. If a custom attribute for " last name furigana " already exists, please refer to 3.7.3 and replace it with a custom attribute ID for each merchant.

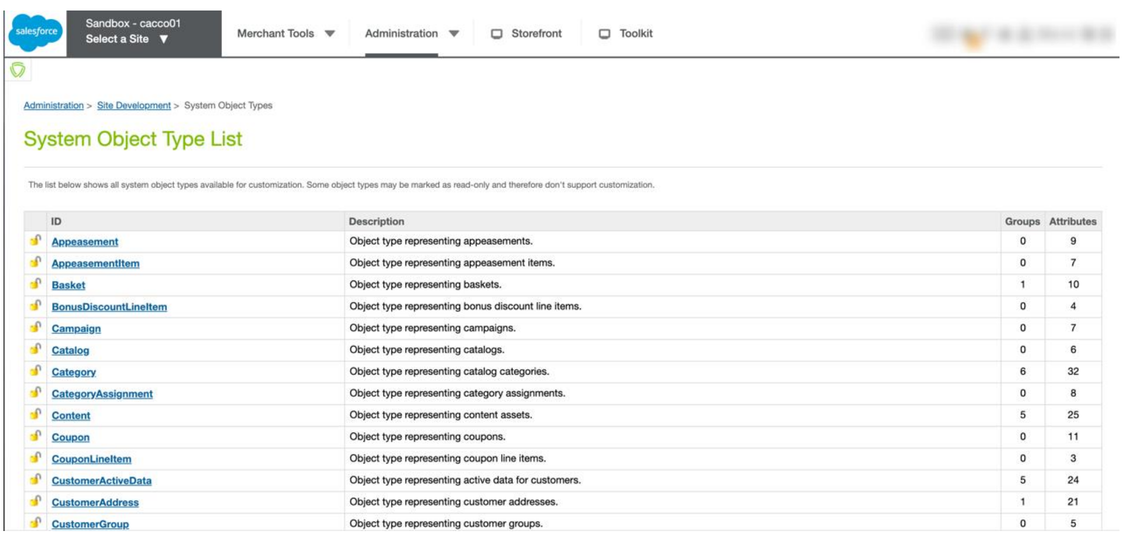
3.7.2 Add Custom Attributes

When using the “delivery mobile phone number”, it is as follows.

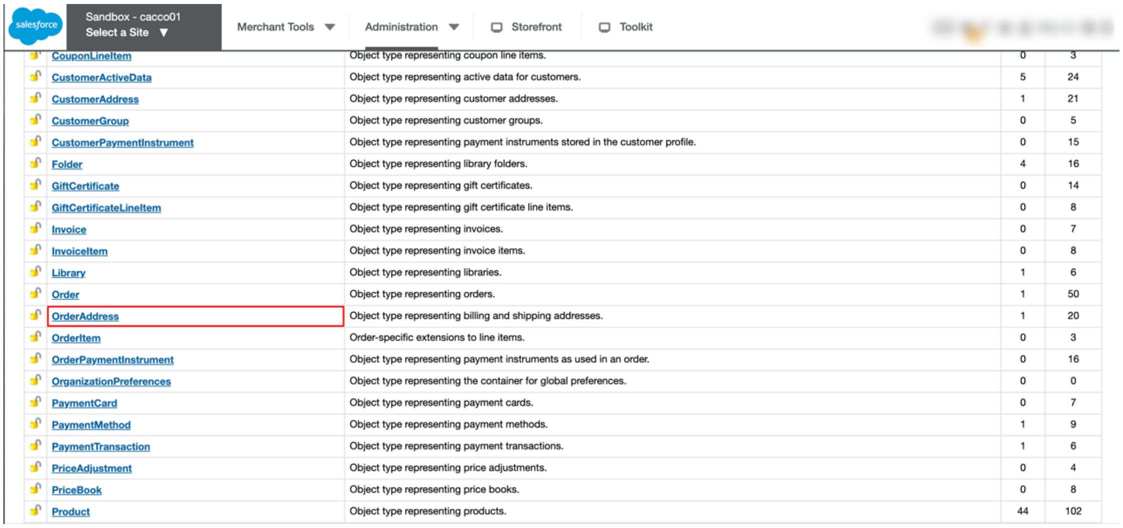
In Business Manager, go to **Administration > Site Development > System Object Types**.



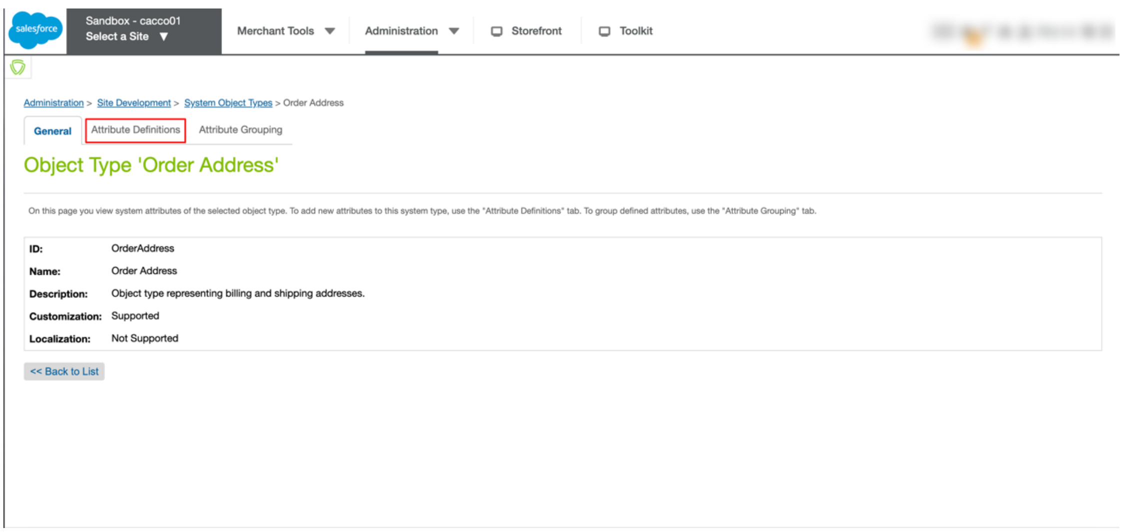
A list of System Objects Types is displayed.



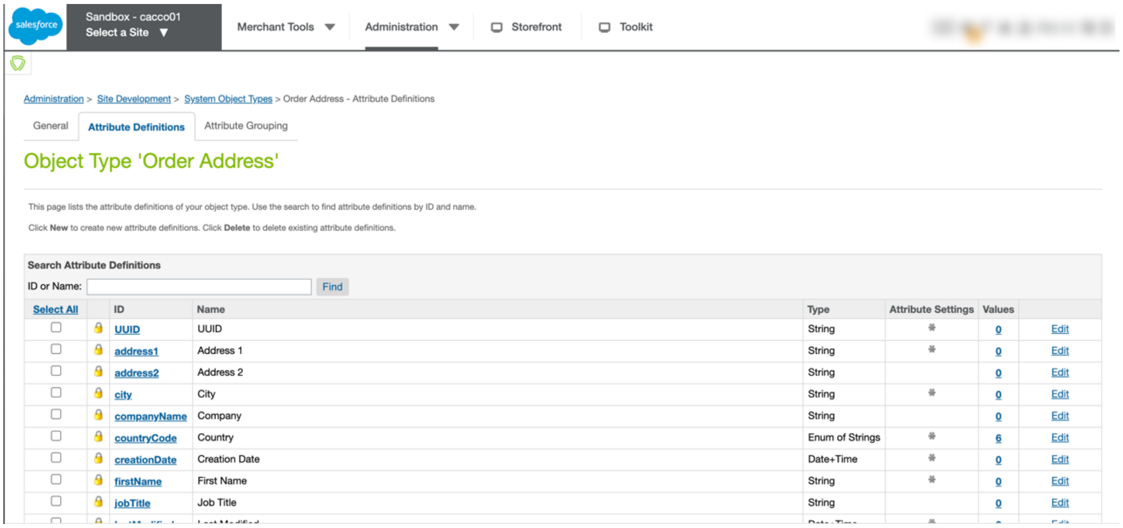
Click “**OrderAddress**”.



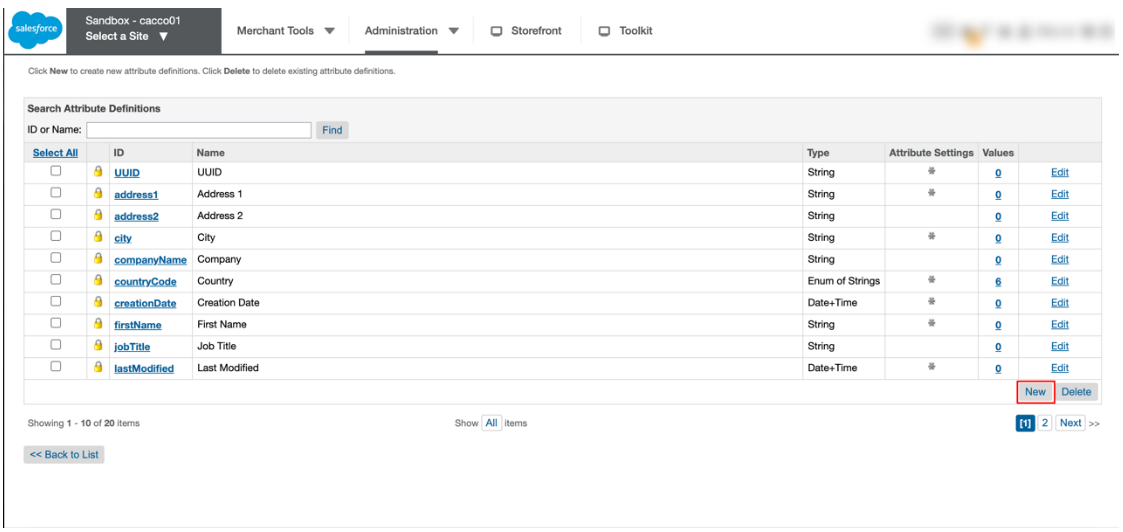
Click “**Attribute Definitions**” tab.



It will be displayed as follows.



Click the “New” button.



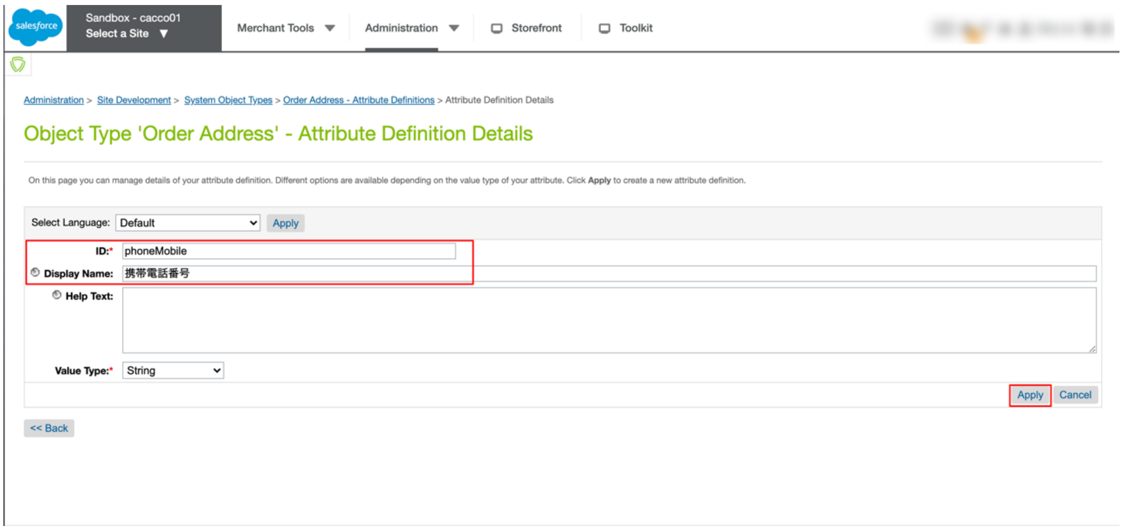
Enter the “ID”, “Display Name”, and “Value Type” of the attribute to be added, and click the “Apply” button.

For example, set as follows.

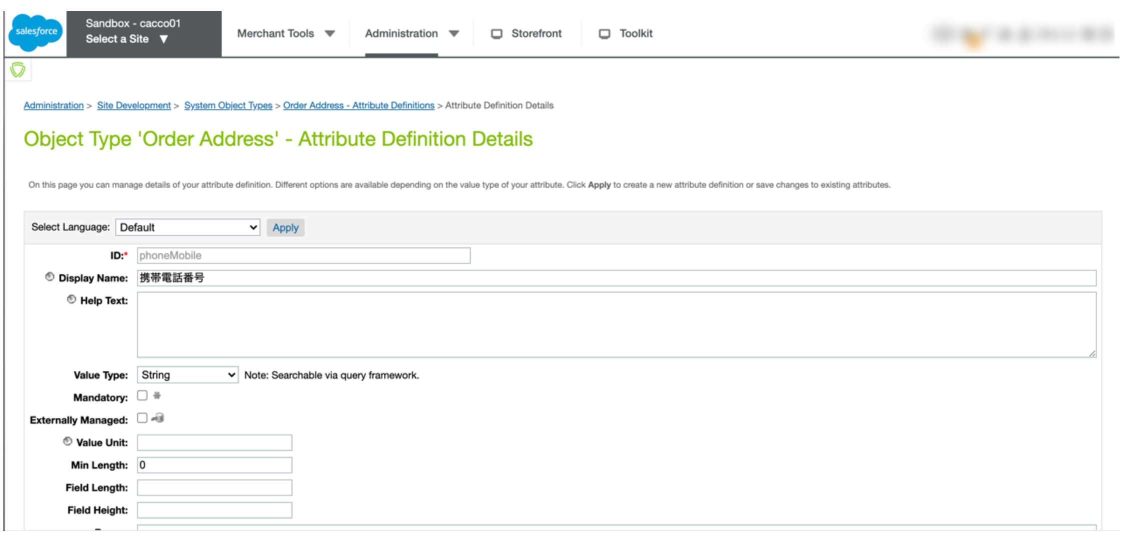
ID: phoneMobile

Display Name: 携帯電話番号

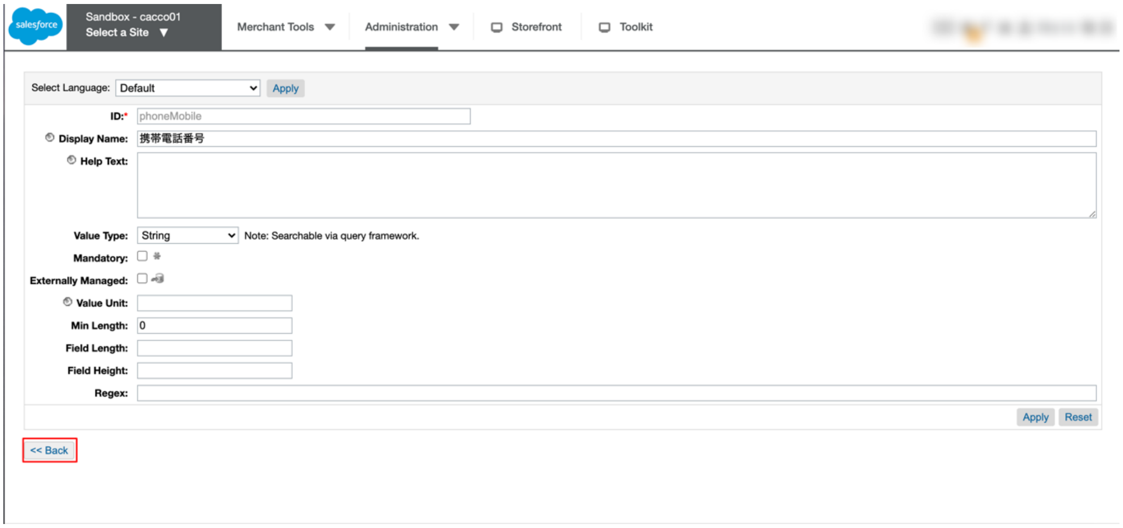
Value Type: String



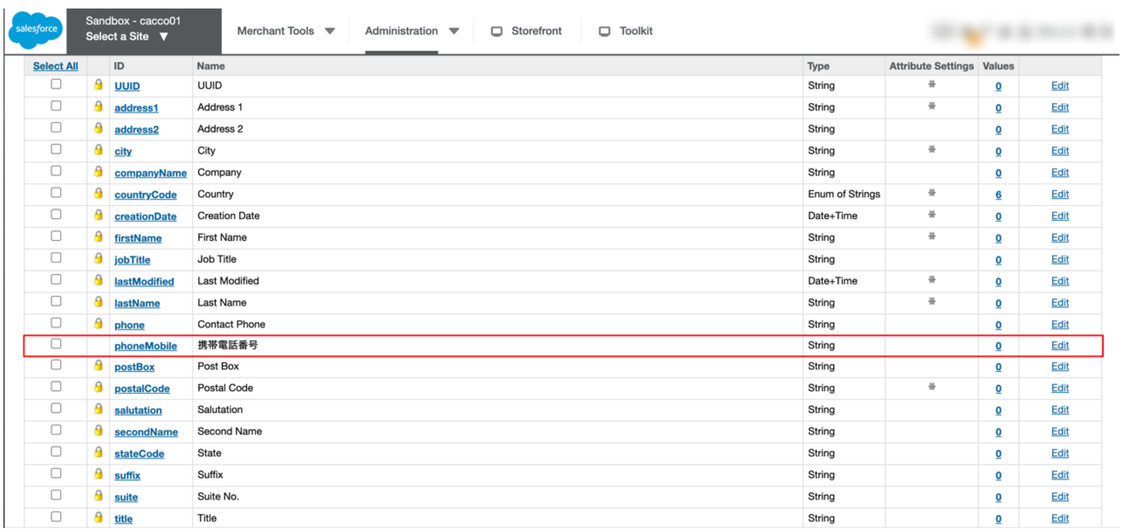
After click the “Apply” button, the following will be displayed.



Click the “Back” button.



Check the added custom attribute.



Select All	ID	Name	Type	Attribute Settings	Values	
	UUID	UUID	String		0	Edit
	address1	Address 1	String		0	Edit
	address2	Address 2	String		0	Edit
	city	City	String		0	Edit
	companyName	Company	String		0	Edit
	countryCode	Country	Enum of Strings		6	Edit
	creationDate	Creation Date	Date+Time		0	Edit
	firstName	First Name	String		0	Edit
	jobTitle	Job Title	String		0	Edit
	lastModified	Last Modified	Date+Time		0	Edit
	lastName	Last Name	String		0	Edit
	phone	Contact Phone	String		0	Edit
	phoneMobile	携帯電話番号	String		0	Edit
	postBox	Post Box	String		0	Edit
	postalCode	Postal Code	String		0	Edit
	salutation	Salutation	String		0	Edit
	secondName	Second Name	String		0	Edit
	stateCode	State	String		0	Edit
	suffix	Suffix	String		0	Edit
	suite	Suite No.	String		0	Edit
	title	Title	String		0	Edit

3.7.3 Code Modification

When adding a parameter to the O-PLUX event registration API,
customize it by referring to the sample code below.

Note : It uses the custom attribute “phoneMobile” created in Step 3.7.2.
Please modify according to the environment.

- Customize **int_oplux/cartridge/scripts/utils/eventHelpers.js**

```
function getObjectForApiRegisterEvent(basketOrOrder, normalizedNames, extraRaw

  var constants = require('*/cartridge/scripts/utils/constants'
  var paymentMethodId
  if (basketOrOrder.paymentInstruments.empty) {
      // Get payment method from page when calling from Storefront
      var server = require('server');
      var paymentForm = server.forms.getForm('billing');
      paymentMethodId = paymentForm.paymentMethod.htmlValue;
   }
  …
  …
  If (basketOrOrder.billingAddress) {
    buyerObj = {
  …
  …
      correctReading: normalizedNames && normalizedNames.buyer &&
normalizedNames.buyer.lastName &&
normalizedNames.buyer.lastName.correctReading ? ‘1’ : ‘0’,
    correctReadingMatchCount: normalizedNames && normalizedNames.buyer &&
normalizedNames.buyer.lastName &&
normalizedNames.buyer.lastName.correctReadingMatchCount || ‘’
```

```
  }
  …
  …
  If (shipment) {
    deliveryObj = {
  …
  …
      correctReading: normalizedNames && normalizedNames.delivery &&
normalizedNames.delivery.lastName &&
normalizedNames.delivery.lastName.correctReading ? ‘1’ : ‘0’,
    correctReadingMatchCount: normalizedNames && normalizedNames.delivery &&
normalizedNames.delivery.lastName &&
normalizedNames.delivery.lastName.correctReadingMatchCount || ‘’
  }
  …
  …
  If (!empty(cardNumber)) {
    eventRequestObj.telegram.event.ec.settle.credit_card = {
      bincode: cardNumber.substr(0, 8) // クレジットカード番号の上 8 桁
    };
  }
eventObj = merge(eventObj, eventRequestObj);
  if(extraRaw){
      eventObj = merge(eventObj, extraRaw);
  }

  return eventObj;
}
```

This cartridge is implemented as link " last name furigana " to the O-PLUX name normalization API as a request parameter, but the value used for linkage is the "oplux_last_name_kana" custom attribute of the CustomerAddress object for convenience. If you would like to use a different attribute ID for " last name furigana " linked to O-PLUX in the merchant SFCC environment, please update "oplux_last_name_kana" in the code below to the attribute ID corresponding to " last name furigana " in the merchant's SFCC environment.

- Customize int_oplux/cartridge/scripts/utils/opluxCallHelpers.js

```
function getNormalizedNames(order) {
    // Name Normalize
    var buyerFirstName = order.billingAddress.firstName;
    var buyerLastName = order.billingAddress.lastName;
    var buyerLastNameKana = order.billingAddress.custom.oplux_last_name_kana;
    var deliveryFirstName = order.defaultShipment.shippingAddress.firstName;
    var deliveryLastName = order.defaultShipment.shippingAddress.lastName;
    var deliveryLastNameKana =
     order.defaultShipment.shippingAddress.custom.oplux_last_name_kana;
    var normalizedNames = {};

    var buyer = OpluxServices.getNormalizedName(buyerFirstName, buyerLastName,
     buyerLastNameKana).getObject();
    var delivery = OpluxServices.getNormalizedName(deliveryFirstName,
     deliveryLastName, deliveryLastNameKana).getObject();

    if (buyer) {
        normalizedNames.buyer = JSON.parse(buyer).response;
    }
    if (delivery) {
        normalizedNames.delivery = JSON.parse(delivery).response;
    }
```

```
    return normalizedNames;
  }
```

Since "oplux_last_name_kana" is also used in the test code, please modify the code below to the merchant's custom attribute ID as well.

- Customize test/mock/int_oplux/cartridge/scripts/utils/eventHelpers.js

```
basket: {
  billingAddress: {
    (省略)
    custom: {
      oplux_last_name_kana: 'ヤマダ'
    },
    postalCode: '980-0022',
    stateCode: 'テスト県',
    phone: 'o12-3456-789'
  },
  customerEmail: 'test.co.jp',
  defaultShipment: {
    shippingAddress: {
      (省略)
      custom: {
        oplux_last_name_kana: 'ヤマダ'
      },
      postalCode: '980-0022',
      stateCode: 'テスト県',
      phone: 'o12-3456-789'
    }
  },
…
```

- Customize test/unit/int_oplux/cartridge/scripts/utils/opluxCallHelpers.js

```
var successResponseNameObject = {
   billingAddress: {
      custom: {
         oplux_last_name_kana: 'ヤマダ'
      },
      firstName: '山田',
      lastName: '太郎'
   },
   defaultShipment: {
      shippingAddress: {
         custom: {
            oplux_last_name_kana: 'ヤマダ'
         },
         firstName: '山田',
         lastName: '太郎'
      }
   },
   …
```

3.8 Customize Content Assets

3.8.1 opluxEventNG

Content assets that are displayed on the storefront when the order status becomes "FAILED" as a result of fraud detection.

Conditions:

- Automatic examination result "NG"

OpluxConfigs "oplux_order_status_for_api_error_limit" setting is "FAILED":

- No API response
- API response exists, processing result "20:error"

OpluxConfigs "oplux_order_status_for_review_hold" setting is "FAILED":

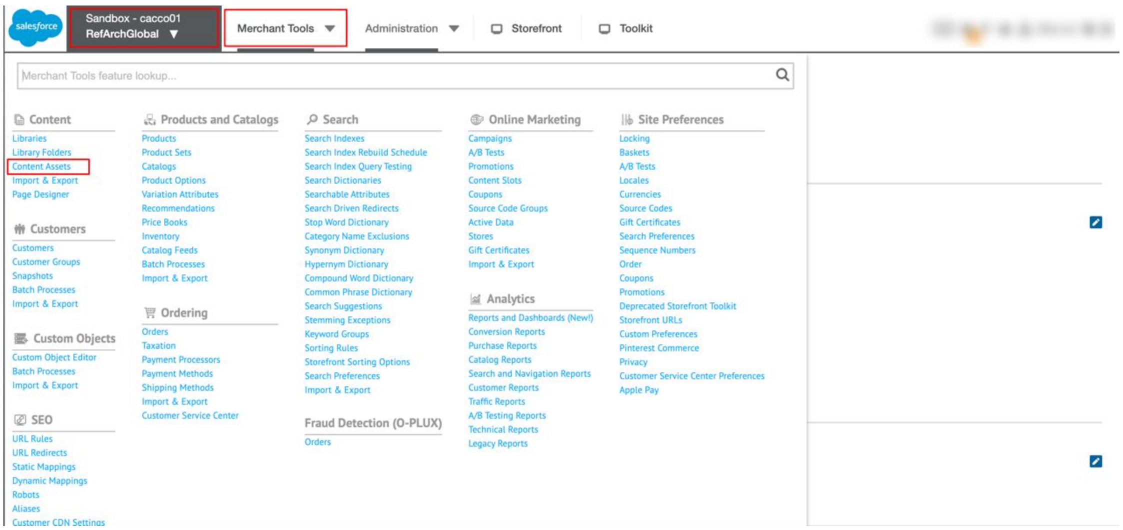
- Automatic examination result 「REVIEW」
- Automatic examination result 「HOLD」

3.8.1.1 Check Content Assets

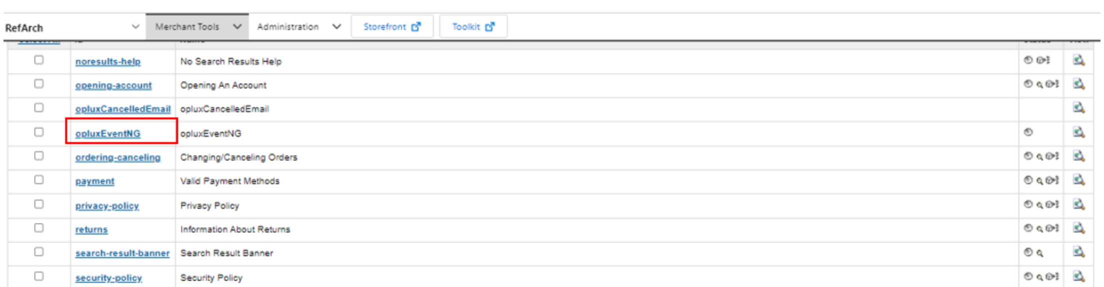
Check the content assets added in "3.2.3 Site Import ".

Select the site to which the content asset was added in Business Manager.

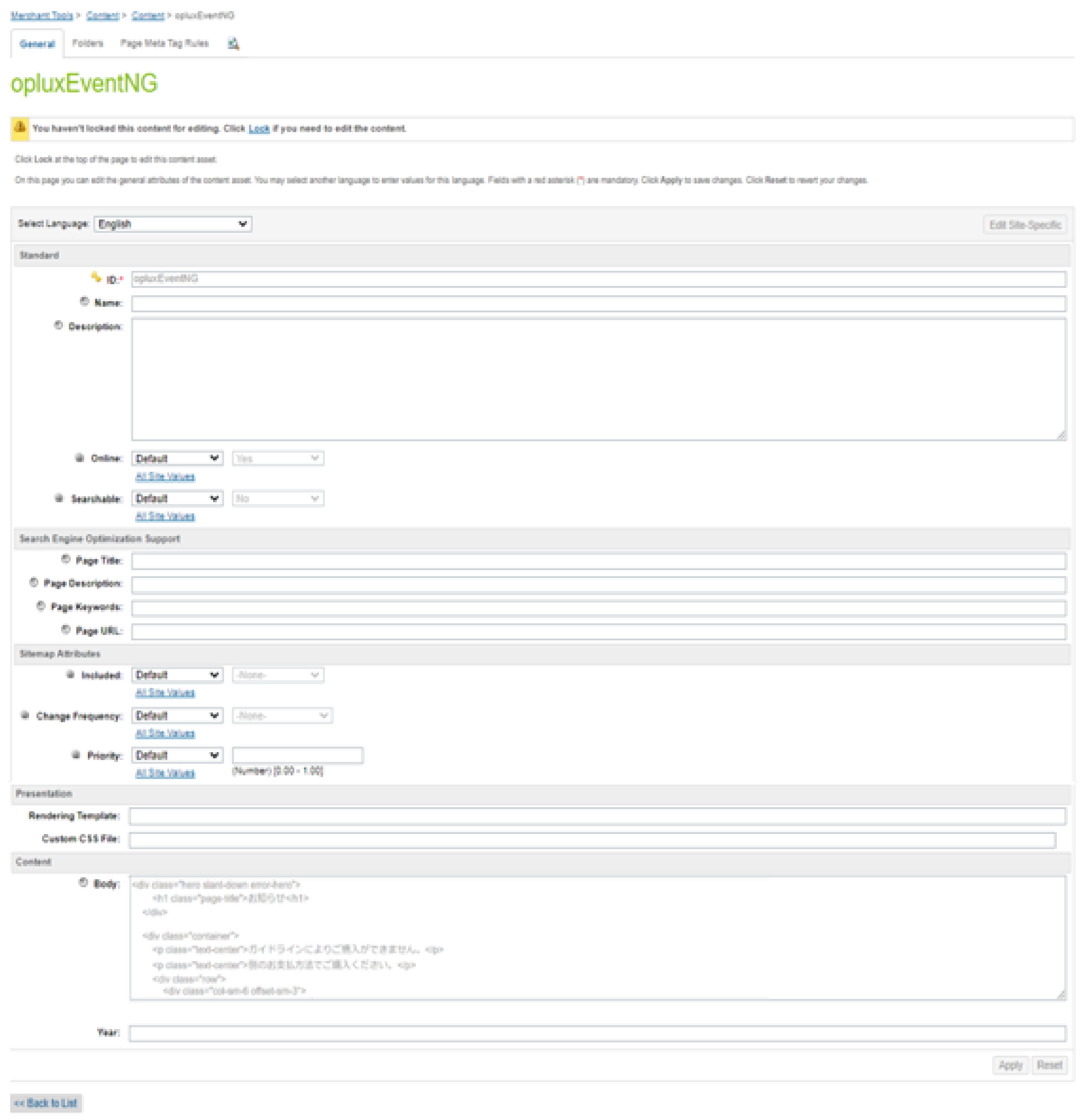
Then go to **Merchant Tools > Content > Content Assets**.



Select [opluxEventNG].



It will appear as follows.

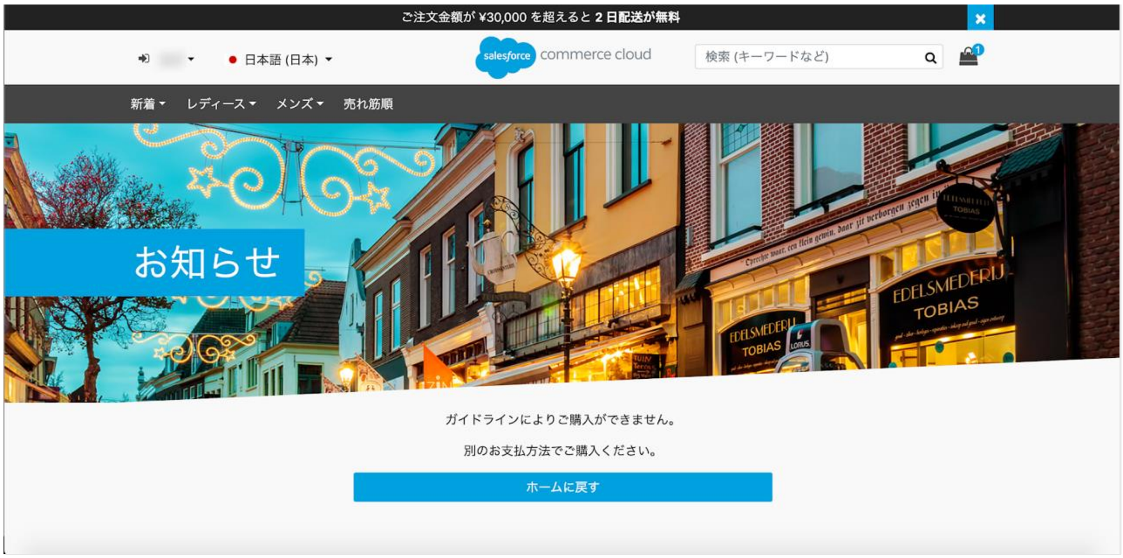


The following information is entered in the Body field.

```
<div class="hero slant-down error-hero">
    <h1 class="page-title">お知らせ</h1>
</div>

<div class="container">
    <p class="text-center">ガイドラインによりご購入ができません。</p>
    <p class="text-center">別のお支払方法でご購入ください。</p>
    <div class="row">
        <div class="col-sm-6 offset-sm-3">
            <a href="$httpsUrl('Home-Show')$" class="btn btn-primary btn-block error
continue-shopping" role="button" aria-pressed="true">
                ホームに戻す
            </a>
        </div>
    </div>
</div>
```

3.8.1.2 Sample Error Page

If the order status is “FAILED” as a result of fraud detection, it will be displayed as follows.



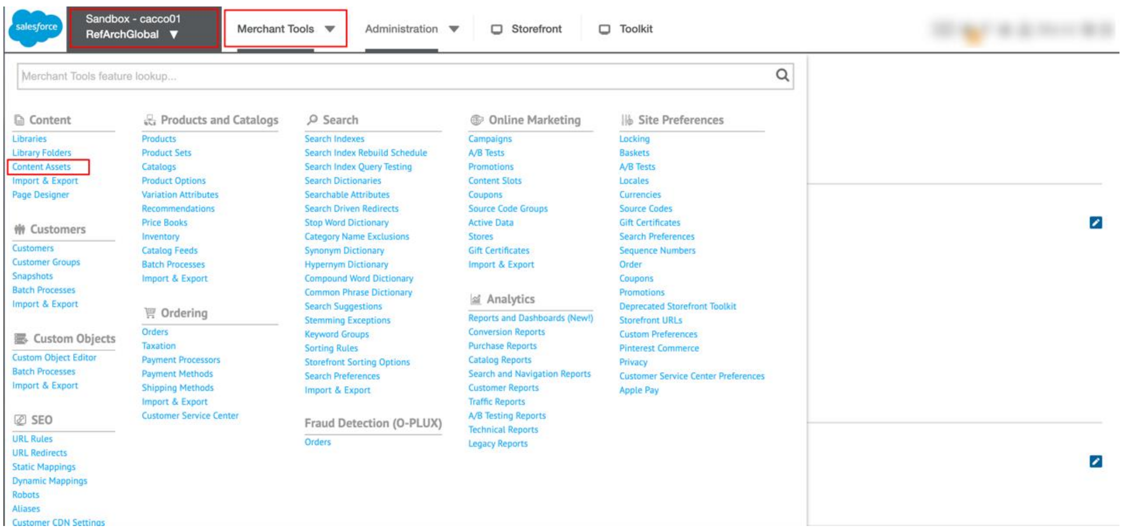
3.8.2 opluxCancelledEmail

This is a content asset for sending a cancellation email to the purchaser when the result of re-examination / Manual examination is “NG”.

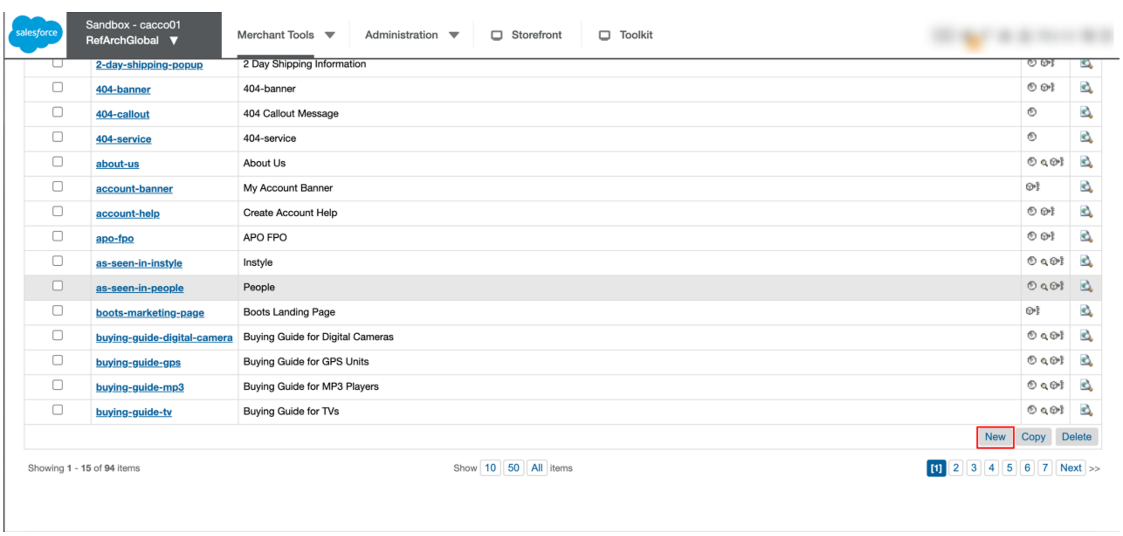
3.8.2.1 Add Content Assets

Select the site which you want to add content assets in Business Manager.

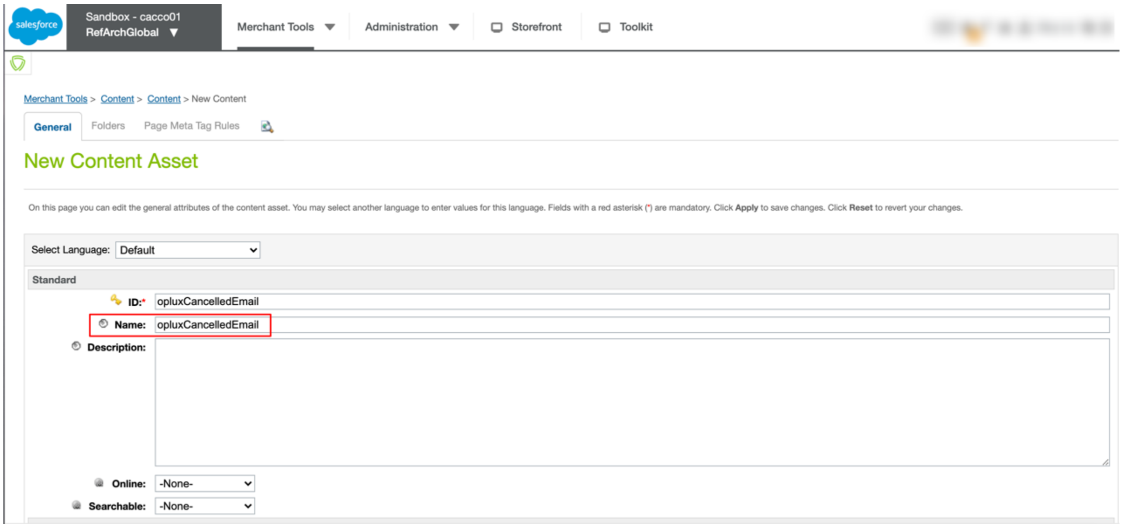
Then go to **Merchant Tools > Content > Content Assets**.



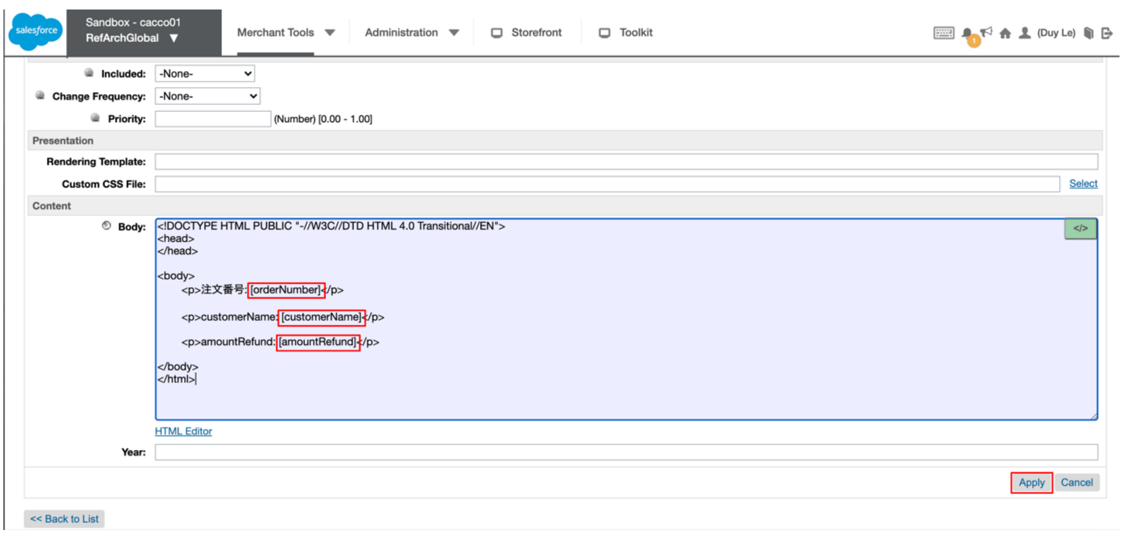
Click the “New” button.



Enter value in “ID” and “Name” fields. The “Name” value will be the title of email.



As an example, the content of the email is the “Order Number”, “Customer Name”, and “Amount Refund”.



3.8.2.2 Code Modification

Please refer to the sample code below for customization.

Note : The example uses “orderNumber”, “customerName”, and “amountRefund” in step 3.8.2.1. Please modify according to the environment.

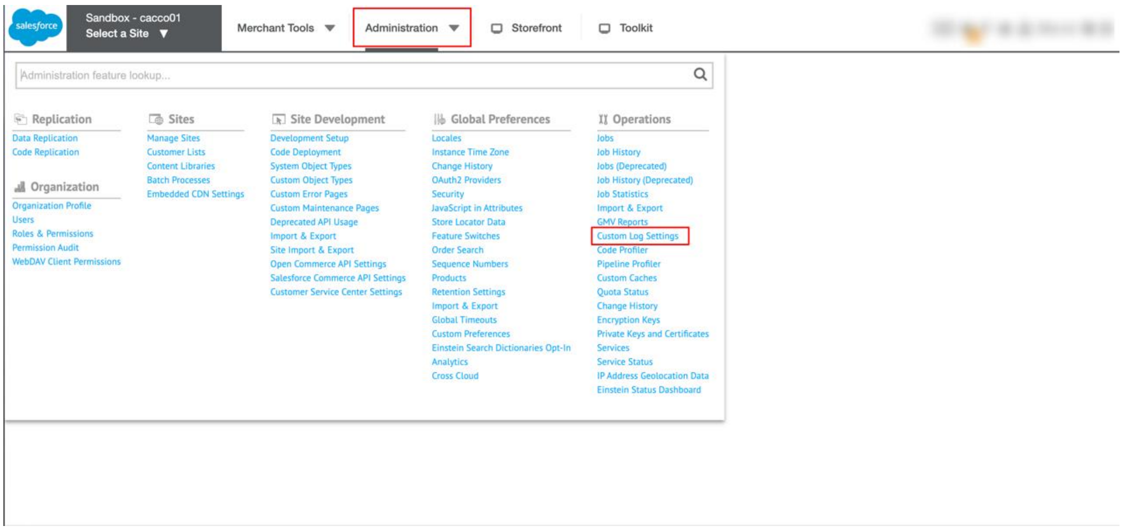
- Customize **int_oplux/cartridge/scripts/utils/mailHelpers.js**

```
/**
* Send Oplux email notification
* @param {dw.order.Order} order - The current order
* @param {string} subject The email subject
* @returns {void}
*/
function sendCancelEmail(order, subject) {
  if (order && !order.custom.oplux_customer_notification_mail_sent_flag) {
     var Transaction = require('dw/system/Transaction');
     var email = order.customerEmail;
     var contentData = {
        customerName: order.getCustomerName(),
        orderNumber: order.getOrderNo(),
        amountRefund: order.getTotalGrossPrice().getValue()
     };

     sendContentEmail(email, 'opluxCancelledEmail', contentData, subject);

     Transaction.wrap(function () {
        order.custom.oplux_customer_notification_mail_sent_flag = true;
     });
  }
}
```

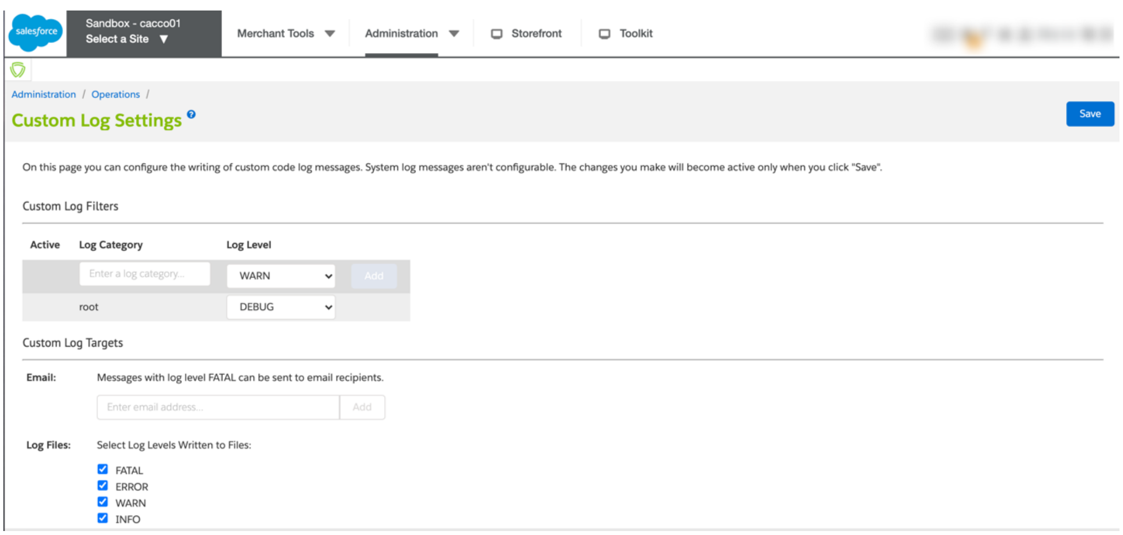
3.9 Log Settings

3.9.1 Output

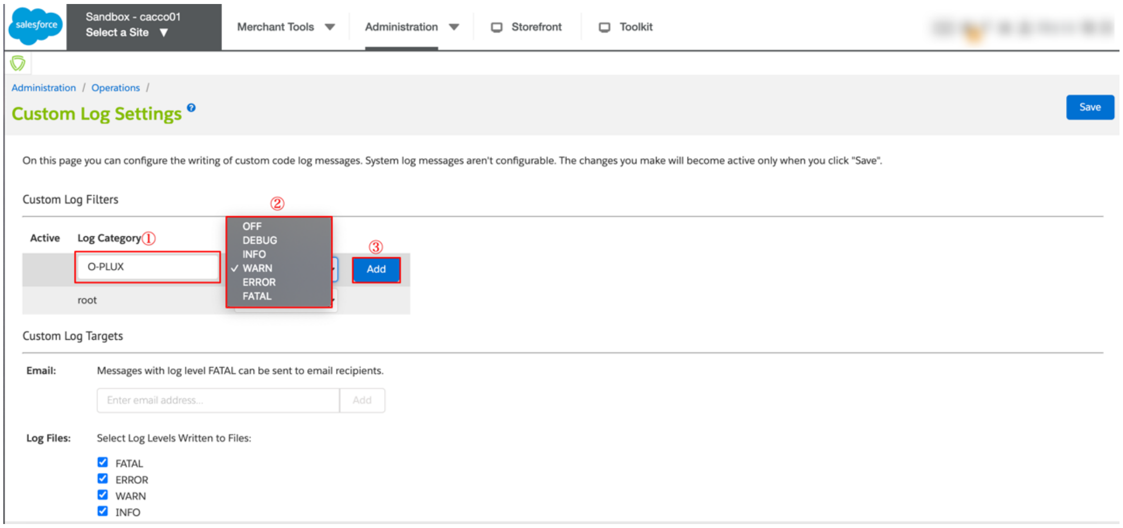
In Business Manager, go to **Administration > Operations > Custom Log Settings**.



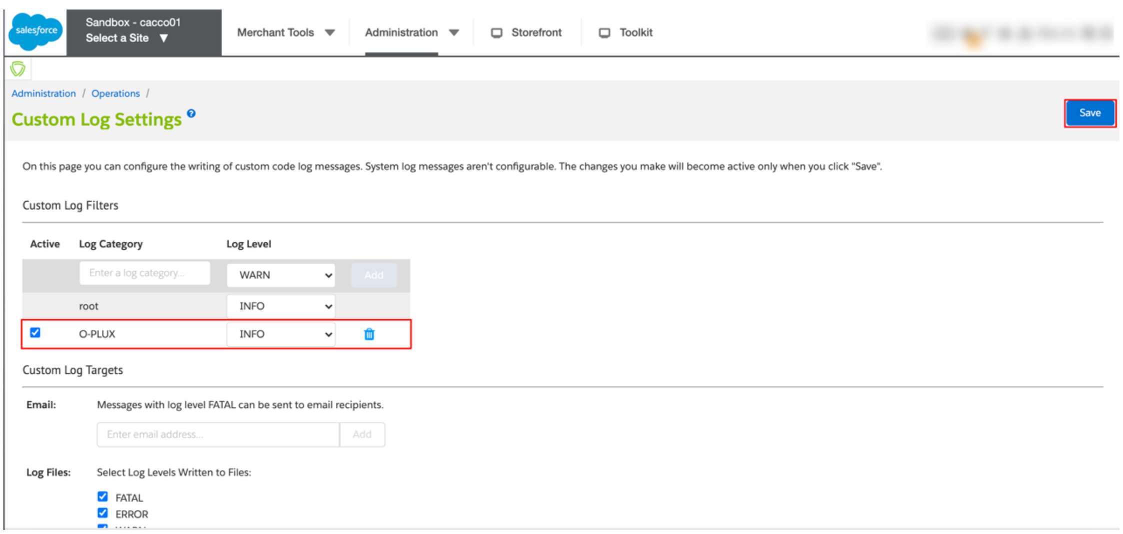
It will be displayed as follows.



Add custom log filter settings. Enter “**O-PLUX**” in the log category, set the log level, and then click the “Add” button.

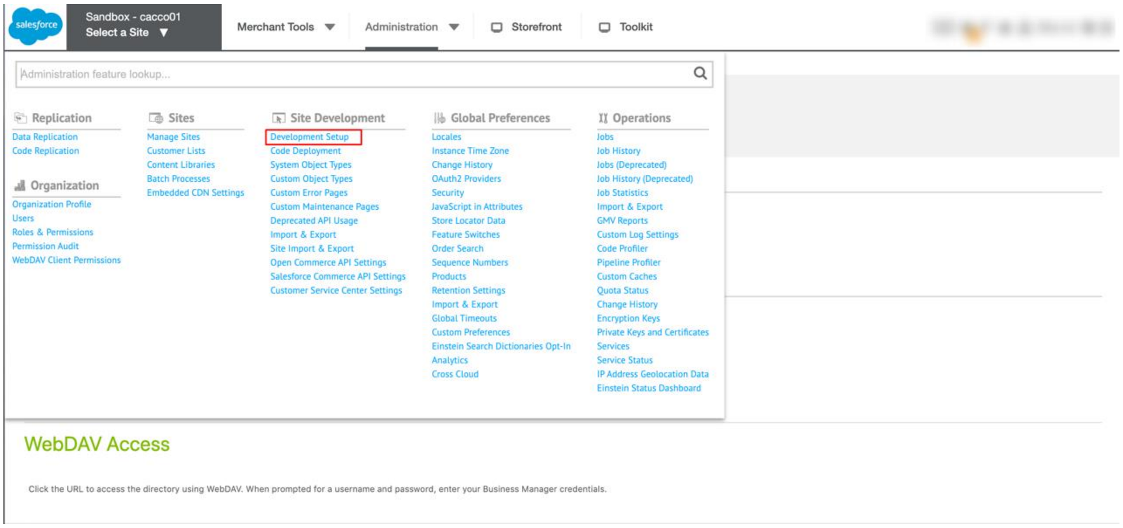


Click the “Save” button to save the custom log settings.

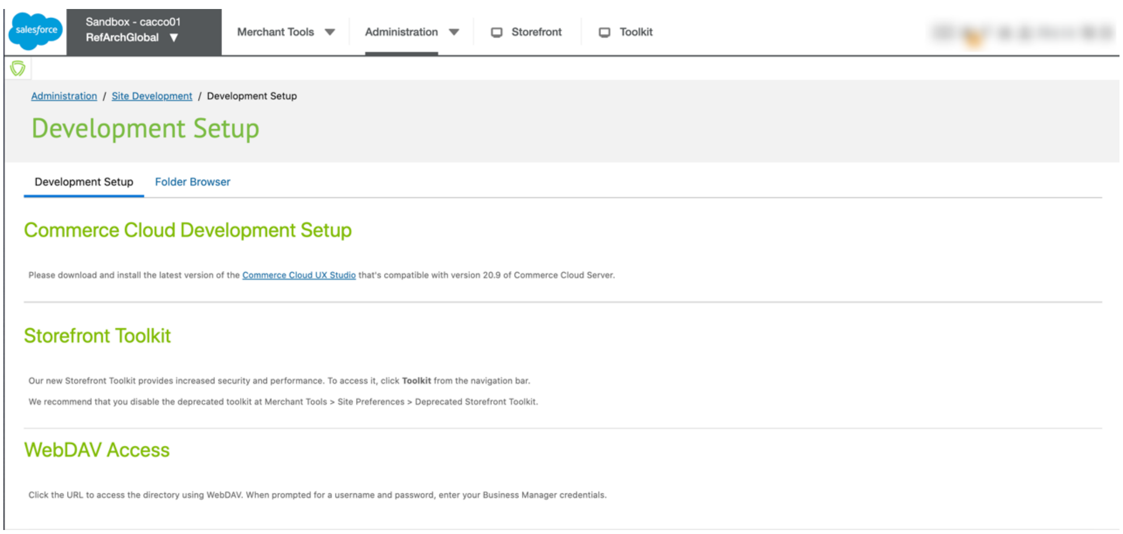


3.9.2 Check log files

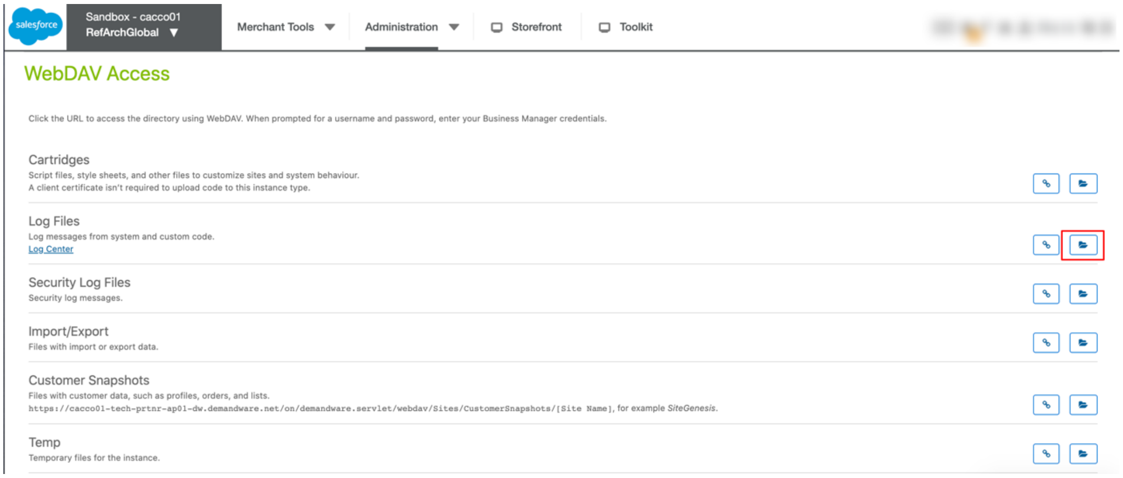
In Business Manager, **Administration > Site Development > Development Setup**.



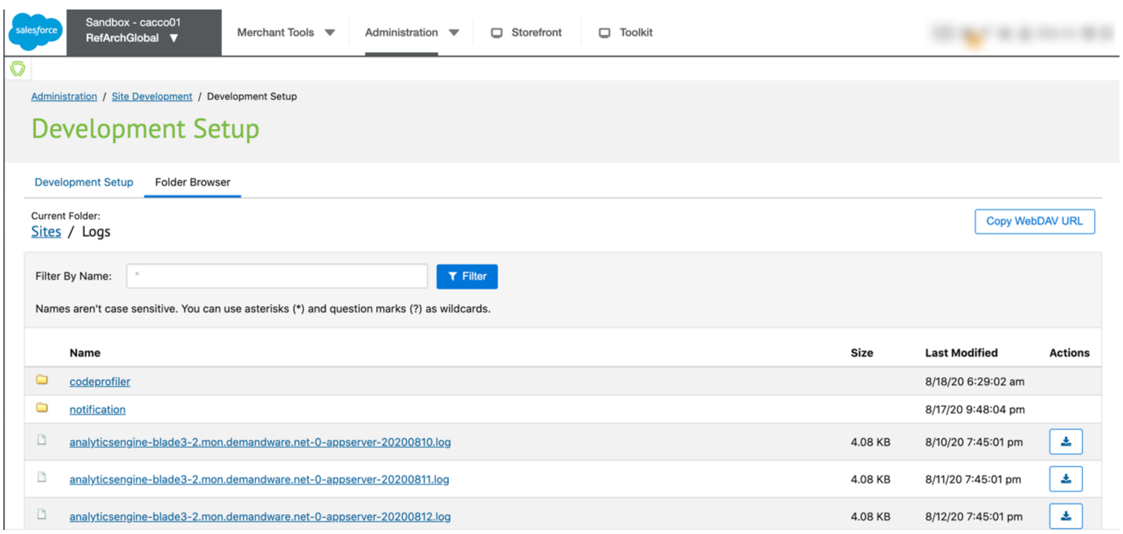
It will be displayed as follows.



Click the URL of the Log Files for WebDAV Access.

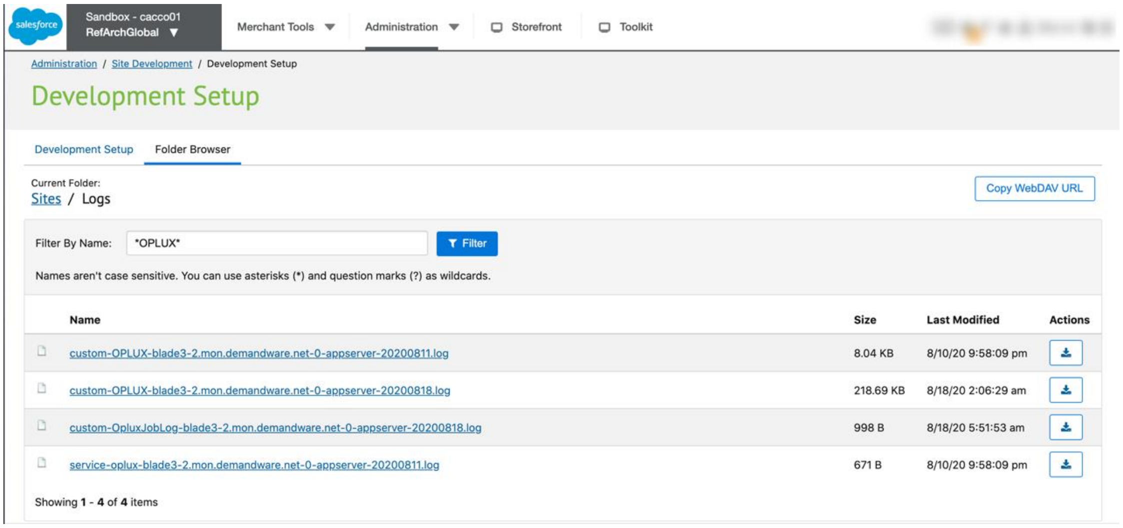


It will be displayed as follows.



There are two types of O-PLUX log files.

- custom-OPLUX-xxxx.log
- custom-OpluxJobLog-xxxx.log



4. Troubleshooting

If the OPLUX service is not functioning, one configuration "oplux_order_status_for_api_error_limit (handling order status when an event registration API error occurs)" must be provided to handle order status. The default value for this setting is to abort the automatic review process and proceed to the next process.

If you wish to change the setting, please follow the steps below.

1. Select the site in Business Manager where you want to use the fraud detection service.
2. Navigate to **Merchant Tools > Site Preferences > Custom Preferences > OpluxConfigs.**
3. Select and set any status on error for oplux_order_status_for_api_error_limit.