

Payeezy JS Integration Guide

April 2015

Contents

Quickly integrate Payeezy JS from First Data into your web applications	3
Minimum technical requirements	3
Payeezy.js initialization	4
Downloadable information	8
Submitting/Generating transaction (method of payments) with example	8
Additional capabilities API	8

Quickly integrate Payeezy JS from First Data into your web applications

If you want to enable secure and convenient payments to your payment applications, this guide will get you up and running quickly. Payeezy handles all the heavy lifting of the complex tokenization and protects your customers' transactions. It is simple to create a developer test account and apply for a merchant account through our developer portal.

PayeezySM, from First Data, is part of the [Small Business Solutions Suite](#), which includes CloverTM, InsighticsSM, PerkaTM and TransArmor®. The [Payeezy eCommerce Solution](#) empowers SMBs to expand their horizons and easily grow their business online or via mobile by reaching new customers no matter where they are.

This documentation refers to the Payeezy JS integration guide included within the overall Payeezy eCommerce Solution. Henceforth, all references to Payeezy are in relation to Payeezy APIs.

Minimum technical requirements

Following are the minimum requirements for Payeezy.js integration

1. You will need TransArmor enabled on your merchant account to do token based transactions. Contact your merchant account representative for more details on how to enable this.
2. For any questions please contact us at [1-855-448-3493](tel:1-855-448-3493). You can also email us at support@payeezy.com
3. Compatible browsers for payeezy.js (any one):

Browser	Version
Firefox	v33.0
Chrome	v39.x
Internet Explorer	IE6+
Safari	v5.1.7

4. Further information can be found in the following [integration document](#)

https://github.com/payeezy/payeezy_js/raw/master/payeezy_js_guide/PayeezyJS_Entitlements_details0412015.pdf

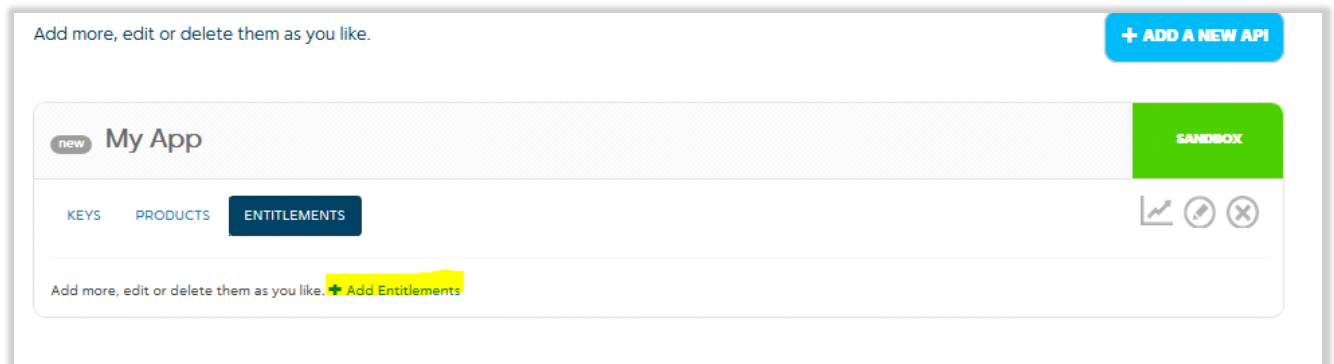
5. Get started: Explore our Payeezy site anonymously! If you like what you see, visit our registration pages to [sign up for a test account](#) and begin the integration process!

Explore our site <http://developer.payeezy.com> anonymously if you like; you'll even be able to test some of the code samples we provide on the API Docs & Sandbox tab. Once you [create an account](#) with us, you'll receive a unique API Key that allows you to generate transactions against a test account on your My Merchants tab. Fully register and certify your solution, and you'll be able to begin bringing merchants on board and generating money!

Payeezy.js initialization

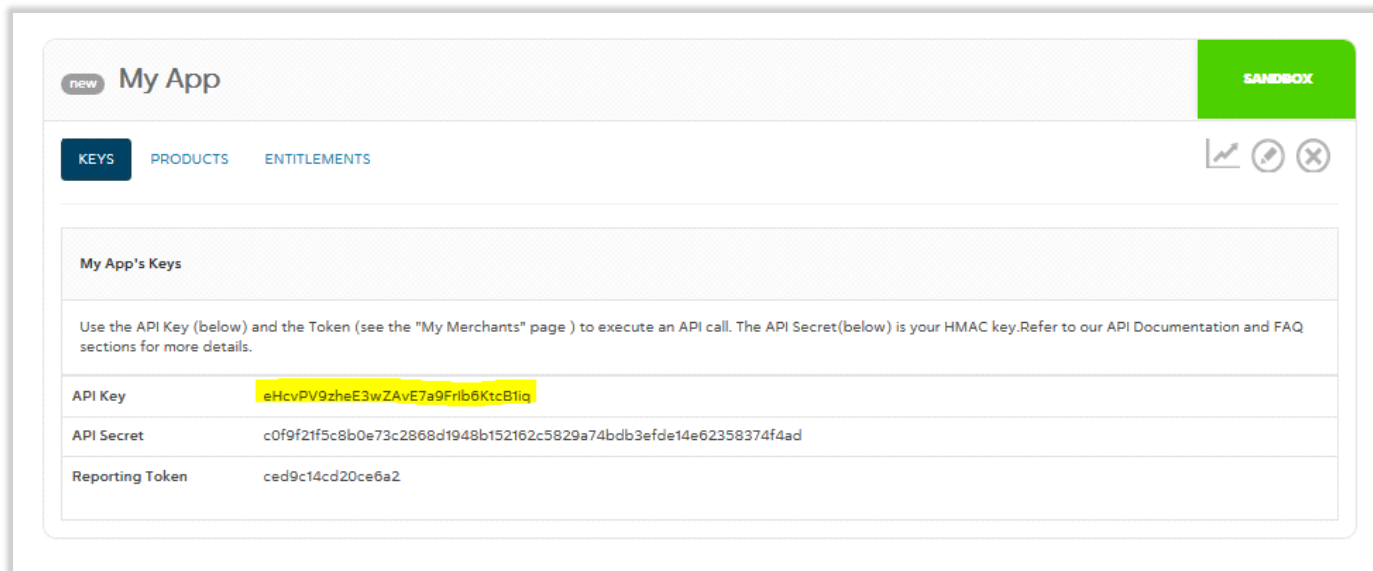
Step by step integration

1. Look here for [prerequisites](#)
https://github.com/payeezy/get_started_with_payeezy/raw/master/get_started_with_payeezy042015.pdf get an overview of our developer portal.
2. Developer portal setup:
Your developer account on developer.payeezy.com needs to be provisioned for Payeezy JS. On the developer portal, select the API you want to entitle, and add Entitlements using the Entitlements Tab



3. Once you've picked the merchant to enable, we will create a unique merchant identifier for you. You will need the API Key and the merchant identifier (highlighted below) to initialize payeezy js.





4. Initializing the Payeezy JS :

Payeezy JS implementation can be initialized by placing the `<script>` tag inside the `<head>` tag of the Payment page. For security reasons, link directly to payeezy.js hosted on Payeezy over https as shown below; do not make a local copy of the js file.

```
<script src="https://developer.payeezy.com/v1/payeezy.js" type="text/javascript"></script>
```

5. Payeezy js has to be initialized by setting the apiKey and the Merchant Identifier inside the `<script>` tag as shown below. For every merchant, api key and merchant identifier will be unique and the same can be retrieved from the developer portal (sample values shown below)

```
<script>

Payeezy.setApiKey('eHcvPV9zheE3wZAvE7a9Frlb6KtcB1iq');

Payeezy.setMerchantIdentifier('5edf46316f6b');

</script>
```

6. Setting up Payment Information Page:

- a. Payeezy.js has to be initialized for setting up the Payment information page, all Personally Identifiable Information (PII) fields on the form should have an attribute "payeezy-data". It is important that these fields do not have the "name" attribute. By not having the "name" attribute, the browser doesn't send that information back to the merchant server. Please look at the highlighted attribute names that are required to generate the payment token. Rest all the fields that are not PII will be submitted to the Merchant server.

```
<form action="{call to merchant server}" method="post" id="payment -info-form" >

<span class="payment-errors"></span>

<div class="row">

<label>Card Type : </label>

<select payeezy-data="card_type">

<option value="visa">Visa</option>
```

7. Creating the Payment Token: Below code snippet can be used to capture the submit event and then create the payment token using the Payeezy JS implementation. It's not required to use jQuery, the same functionality can be accomplished with native javascript.

```
<option value="purchase" >Purchase</option>

<option value="authorize" selected="selected" >Authorize</option>

</select>

</div>

<div class="row">

<label>Amount : </label>

<input type="text" name="amount" id="amount" value="101"/>

</div>

<button type="submit">Make Payment</button>

</form>
```

“responseHandler” is a merchant defined JS function, which will be called when the creation of payment token is successful. Example of this function is provided in the next section.

```
<option value="05">05</option>

<option value="06">06</option>

<option value="07">07</option>

<option value="08">08</option>

<option value="09">09</option>

<option value="10">10</option>

<option value="11">11</option>

<option value="12" selected>12</option>

</select>

<select payeezy-data="exp_year">

<option value="14" selected>2014</option>

<option value="15">2015</option>

<option value="16">2016</option>

<option value="17">2017</option>

<option value="18">2018</option>

</select>
```

8. Sending the Payment Information to Merchant Server:

“responseHandler” method take two arguments status and response(a javascript object). If the payment token creation is successful, then a status code of “201” is returned. Anything other than a “201”, the error message can be got from the response object. Highlighted code shows how to get the error and the token value(if successful)

```
var responseHandler = function(status, response)

{

    var $form = $('#payment-info-form');

    if (status != 201) {

        if (response.error) {

            var errorMessages = response.error.messages;

            var allErrors = '';

            for (i=0; i<errorMessages.length;i++) {

                allErrors = allErrors + errorMessages[i].description;
```

The token value will be put in a hidden field (payeezyToken in this case), and then the form will be submitted to the Merchant server for making the payment with the “Token” instead of credit card information.

9. Make a Payment (server side) using the payment token:

Please refer to the Developer portal documentation on how to make a payment ("authorize" or "purchase") with the token.

10. Deploy the Payeezy.js and dependency java script files in the specified folder structure in your website JavaScript deployment folder.

Downloadable information

The sample code for payeezy.js can be found on GitHub. Following is the GitHub link for the sample code.

<https://github.com/payeezy/payeezy>

<https://developer.payeezy.com/sdk-downloads>

Submitting/Generating transaction (method of payments) with example

Payeezy supports the following method of payments

- Credit Card Payments
- PayPal Transactions
- Gift Card (via ValueLink) Transactions
- eCheck (via TeleCheck) Transactions

For API processing details, click here [Method of Payments](#)

Additional capabilities API

- Partner Reporting API - Use our powerful query engine to retrieve payment records. Supports complex filtering, sorting, pagination and more. This is exclusively for Third Party Partners and applicable for a **live** environment only.

For Reporting API processing details, click here https://developer-ga.payeezy.com/payeezy_ref_docs/apis/get/transactions-0