

# IFrame Integration Guide

---

v2.0.0

## Contents

- 1 Contents
  - 1.1 Changelog
- 2 Overview
- 3 Terminology Used
- 4 Testing
- 5 Security
- 6 Transactions
  - 6.1 Request Sample:
  - 6.2 Nodes:
- 7 Rebill Feature
  - 7.1 Request Sample:
  - 7.2 Rebill Nodes:
- 8 Customization Features
- 9 Migrating to PRODUCTION account and live processing
- 10 Notifications [OPTIONAL]
- 11 850- Command Notification
  - 11.1 850 Notification Sample
  - 11.2 850 Notification Data description
  - 11.3 850 Notification Response
  - 11.4 850 Response Data description

---

## Changelog

Version	Date	Description	Author
1.0.0	Jul 08, 2016	Initial Version	David Ramirez
1.1.0	Jan 16, 2017	Added Rebill features	Jorge Villafuerte
1.1.1	Feb 02, 2017	Added customization features	David Ramirez

---

## Overview

The GPN DATA Payment Page is the method by which GPN DATA conducts secure CNP (Card Not Present) transactions when the Merchant either chooses not to provide the Payment Page or the Merchant site does not meet PCI requirements. The target audience of this document is the Programmer or IT Professional who will be integrating the GPN DATA Payment Page into the Merchant's web site.

GPN DATA strictly adheres to the standards set forth by PCI-DSS (Payment Card Industry – Data Security Standard). The intent of the Payment Page is to remove the possibility of the Merchant acquiring CHD (Card Holder Data). Any attempts to circumvent this protection and attempt to acquire CHD (Card Holder Data) are a violation of PCI-DSS standards. To obtain the latest information regarding PCI-DSS please consult:

[https://www.pcisecuritystandards.org/security\\_standards/documents.php](https://www.pcisecuritystandards.org/security_standards/documents.php)

All information contained in this document is the intellectual property of GPN DATA Europe Ltd. and is protected by International Copyrights.

All references in this document to “gateway” or Gateway” directly refer to GPN DATA Europe unless otherwise specified.

## Terminology Used

**Gateway** – The web server, database server and programs associated with Credit Card processing

**Transaction** – Any exchange of date between the Gateway and the Merchant.

**CHD** – Card Holder Data as described by PCI (the Payment Card Industry and CHD are addressed in more detail in the Security section of this document).

**Charge** – A Charge transaction performs an Authorize transaction and immediately follows with an automatic Capture transaction

**Chargeback** – Chargebacks are created when the Cardholder informs the Issuing Bank that there is a problem with the goods or services received or not received from the merchant.

**3DS** – 3DS is a generic name for programs from several of the card companies. The formal names are: Verified by Visa, Secure Code by Mastercard and JSecure from JCB. Transactions processed under 3DS involve the Cardholder being redirected to the Issuing Bank for transaction Authorization. Transactions processed as 3DS provide some levels of protection for the Merchant against Chargebacks resulting from fraudulent charges.

---

## Testing

For each Merchant account, the Merchant will be issued login credentials and URL for the Merchant Back Office on a TEST server. All necessary API documentation and connection information will be located there. The TEST server has the following features to assist in integration:

1. Initially, the Acquiring Bank will be an internal “test bank” and no actual charges will be placed on any card used for testing.
  2. CVV values of 000 to 799 will result in the transaction returning as following:
    - 001 to 249** – emulate non-enrolled card, transaction approved;
    - 250 to 499** – emulate enrolled card (with 3D Secure redirection), Transaction approved after successful ACS authentication;
    - 500** – emulate transaction error
    - 501 to 749** – emulate non-enrolled card, transaction declined;
    - 750 to 799** – emulate enrolled card (with 3D Secure redirection), Transaction declined even after successful ACS authentication.
  3. There is no limit to the number of transactions or the access time for testing.
  4. Avoid issuing Customer names like Test Test or John Dohn, You may use any valid card number (except 4111111111111111). If you need test card numbers, please search the web for “get credit card numbers”.
  5. No transaction fees will be assessed to the Merchant during testing
  6. Once testing is completed with the “test bank”, the Merchant account will be connected to the Acquirer assigned by Sales as indicated by the Merchant Agreement and all further testing will involve actual processing by the Acquirer and all transactions will be treated as actual charges to the card.
  7. Once all testing is completed and Sales has issued instructions, new Merchant Back Office login credentials and URL will be issued, along with a new URL for transmitting API Transactions. The Merchant Account on the TEST server will remain active after the Live Account is operational.
- 

## Security

There are several overriding concerns with regard to security: CHD (Card Holder Data), Transaction Verification and Intrusion Prevention to mention just a few.

GPN DATA does not record or save sensitive CHD. Even the transaction logs are “sanitized” before being written to disk to mask the PAN (Primary Account Number) and the CVV. All GPN DATA procedures with regard to CHD are in adherence to PCI-DSS specifications as mentioned in the Overview section.

All Transactions between merchants and GPN DATA include a checksum. The checksum is created using a Secure Hash Algorithm. The Payment Page uses SHA-1 algorithm and an API key assigned to every merchant by GPN DATA. This API key is never included in the actual transaction, but is used in calculating the checksum. You will find the API key on the “Technical Integration” menu in the Merchant Back Office.

All transactions will contain a checksum; the checksum is calculated by concatenating data from specific fields in the transaction (the appropriate fields are indicated in the API Transaction Tables listed below), the secret API key is then appended to the end of the string and the checksum is calculated using SHA-1 algorithm. Before accepting the transaction request, each side must perform this procedure and compare checksums in order to validate the transaction.

---

## Transactions

### Request Sample:

```

<html>
<head>
    <title>Merchant Site</title>
</head>
<body>
    <h2>Merchant Site</h2>
<button
  type="button" class="gpn-button-sample"
  data-merchant-id="merchant-pretty-name"
    data-amount="2000"
    data-currency="EUR"
    data-company="Merchant CompanyName"
    data-description="Description of Product or Service"
    data-logo="/assets/img/logo.png"
    data-billing-address="true"
  data-order-id="order-reference"
  data-customer-id="customer-account-id"
  data-customer-email="customer@mail.com"
  data-customer-phone="48000000000"
  data-customer-firstname="John"
  data-customer-lastname="Doe"
  data-customer-birthdate="1989-08-30"
  data-customer-addressline="Street 24"
  data-customer-zipcode="12345"
  data-customer-city="Wonderland"
  data-customer-state="MA"
  data-customer-country="POL">
  Button Pay with GPN DATA
  >Pay Button 1
</button>

<script src="https://[payment page domain]/checkout.js"></script>

</body>
</html>

```

## Nodes:

Node	Required	Description
data-merchant-id	Required	Merchant ID will be provided by TechSupport, during initial communication.
data-amount	Required	Amount must be expressed in cents, Example 2000 cents in order to pay \$20.00
data-currency	Required	Currency must be expressed by 3 letters ISO 4217. Example : USD If wished multiple currencies can be shown in a drop down menu by listing separate by command as following "USD,EUR,GBP". Note : Only currencies specified in the Merchant Agreement can be used

data-description	<b>Required</b>	Description of the product or service
data-company	<b>Optional</b>	If not provided GPN Data will be visible at the iFrame checkout
data-logo	<b>Optional</b>	If not provided GPN Data Logo will be visible at the iFrame checkout Image should be hosted at domain, and field should only list the directory of the file or can be imagine can be provided to GPN Data Techsupport for manual upload.
data-billing-address	<b>Optional</b>	Default is set to "false" as the same Customer address will be used for Billing. set to "true" to be included in the form
data-order-id	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Order ID on merchant web site. (if not included, this will be generated by the PaymentPage)
data-customer-id	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer ID on merchant web site. (if not included, it will be empty)
data-customer-email	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer Email address.
data-customer-phone	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer phone number.
data-customer-firstname	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer first name.
data-customer-lastname	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer last name.
data-customer-birthdate	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer last BirthDate. or set to "true" in order to be shown in the form.
data-customer-addressline	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer street address.
data-customer-zipcode	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer zipcode.
data-customer-city	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer city
data-customer-state	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer State two letter ISO
data-customer-country	<b>Optional</b>	To be used for auto population of the iFrame checkout form. Customer State three letter ISO
data-merchant-spec1	<b>Optional</b>	To be used for additional merchant data, which will be received in the Notification API call 850
data-merchant-spec2	<b>Optional</b>	To be used for additional merchant data, which will be received in the Notification API call 850
data-merchant-spec3	<b>Optional</b>	To be used for additional merchant data, which will be received in the Notification API call 850
data-merchant-spec4	<b>Optional</b>	To be used for additional merchant data, which will be received in the Notification API call 850
data-merchant-spec5	<b>Optional</b>	To be used for additional merchant data, which will be received in the Notification API call 850

## Rebill Feature

In order to enable rebill feature a "data-rebill" node with a **TRUE** value must be added on the button data.

### Request Sample:

```
<html>
<head>
    <title>Merchant Site</title>
</head>
<body>
    <h2>Merchant Site</h2>
<button
  type="button" class="gpn-button-sample"
  data-merchant-id="merchant-pretty-name"
    data-amount="2000"
    data-currency="EUR"
    data-company="Merchant CompanyName"
    data-description="Description of Product or Service"
    data-logo="/assets/img/logo.png"
    data-billing-address="true"
  data-order-id="order-reference"
  data-customer-id="customer-account-id"
  data-customer-email="customer@mail.com"
  data-customer-phone="48000000000"
  data-customer-firstname="John"
  data-customer-lastname="Doe"
  data-customer-birthdate="1989-08-30"
  data-customer-addressline="Street 24"
  data-customer-zipcode="12345"
  data-customer-city="Wonderland"
  data-customer-state="MA"
  data-customer-country="POL"

    data-rebill="true"
    data-allow-rebill="false"
    data-rebill-amount="2.33"
    data-rebill-freq="1m"
    data-rebill-count="5"
    data-rebill-start-date="2016-12-29">
  Button Pay with GPN DATA
  >Pay with Rebill
</button>

<script src="https://[payment page domain]/checkout.js"></script>

</body>
</html>
```

**Note**

Rebill execution will have to be done via GPN Data Merchant Back Office

**Rebill Nodes:**

Node	Required	Description
data-rebill	<b>Required</b>	Value must be TRUE to enable rebill feature.
data-allow-rebill	<b>Optional</b>	If TRUE a checkbox will appear on the iFrame and the customer must check it accepting the rebill payment
data-rebill-amount	<b>Optional</b>	Amount to be charged periodically. Must be expressed in a decimal with 2 decimal digits Example \$20.00, \$99.99
data-rebill-freq	<b>Optional</b>	Period of time between charges. Must be expressed {digit}(d w m y) expression d = days, m = month, y = year. Example 30d, 1m.
data-rebill-count	<b>Optional</b>	Number of periodic charges.
data-rebill-start-date	<b>Optional</b>	Date on which rebill should start. Must be expressed in the following format YYYY-mm-dd

## Customization Features

Following option can be added to the iFrame form:

Node	Required	Description
data-auto-open	<b>Optional</b>	If set to "true" iframe page will auto open when using pop-up mode
data-page-container	<b>Optional</b>	If data-page-container = ".payment-form" and it's specified in a component selector, for example class="gpn-payform" Payment page will open in flat mode and not in pop-up mode
data-skip-customerinfo	<b>Optional</b>	If set to "true" iframe page will open in Step2 - Card Data In order to use this option, customer data needs to be sent in the form

**iframe CSS**

```
.carousel {  
  position: relative;  
  height: 452px;  
}  
#modal-payment .card{  
  border-radius: 6px;  
}  
  
#modal-payment a[href],
```

```
#modal-payment .title .text-info{
    color: #87bf3d;
}

#modal-payment .form-control{
    background-color: white;
    padding: 11px 12px;
    height: 45px;
    margin-bottom: 9px;
    color: #333;
    font-size: 16px;
}

#modal-payment .form-control:focus{
    border-color: #87bf3d;
    outline: 0;
    box-shadow: inset 0 1px 1px rgba(0,0,0,.075), 0 0 8px rgba(135, 191,
61, .6);
}

#modal-payment .btn.btn-primary{
    background-color: #87bf3d;
    border: none;
    line-height: 31px;
    box-shadow: 0 3px 0 rgb(102, 146, 45) !important;
    border-radius: 2px;
    font-size: 19px;
    text-transform: uppercase;
    font-weight: normal;
}

#modal-payment .btn-pay-group .td{
    padding-bottom: 4px;
}

#modal-payment .btn-pay-group .back .btn{
    border-color: #777;
    background-color: #777;
    box-shadow: 0 3px 0 #555 !important;
    border-top-right-radius: 0px;
    border-bottom-right-radius: 0px;
}

#modal-payment .btn-pay-group #btn-pay{
    border-top-left-radius: 0px;
    border-bottom-left-radius: 0px;
}

.container .title {
    margin-top: -20px !important;
}
```



```

        margin-left: auto !important;
        margin-right: auto !important;
    }
    .container .title .td.logo {
        width: 288px !important;
        height: 79px !important;
        background: no-repeat center center;
        background-size: 170px 46px;
        max-width: 292px !important;
        background-image: url(https:URL/logo.png);
    }

    .container .title .td.info,
    .container .title .td.logo img {
        display:none !important;
    }
    .double-bounce1, .double-bounce2 {background-color: #87bf3d
!important;}
    .form-group {
        margin-bottom: 10px !important;
    }

        #languages .btn img {
            margin-right: 12px;
            display: none;
        }
    #languages .btn {
        text-align: center;
        width: 100px;
        min-width: 100px;
    }
    #languages .dropdown-menu {
        text-align: center;
        width: 100px;
        min-width: 100px;
    }
    #languages .dropdown-menu a img{
        display: none;
    }

```

```
.footer-info {  
font-size: 13px;  
}
```

---

## Migrating to PRODUCTION account and live processing

Once we receive permission from your Account Manager, you will receive an email with login credentials to your MBO on the PRODUCTION server.

You will be required to change the script src= from <https://ppt.txpmnts.com/checkout.js> to the URL provided by TechSupport on each your buttons.

## Notifications [OPTIONAL]

After each transaction is completed, an Asynchronous Notification message will be sent from the Gateway Transaction Server to the Merchant Transaction Server at the URL provided by the Merchant. While charge transactions can be processed without the Notification, it is highly recommended to add the necessary program logic for handling the Notifications as additional information about the transaction is contained therein. Some API commands, such as Refunds, cannot be transmitted without the additional data contained in the Notifications.

### Note

If (and we stress the word "IF") the cardholder clicks on the link to return to the Merchant at the end of the Payment Page process, the URL will have 2 POST variables attached: (1) id=merchantransid and (2) res=status (APPROVED or DECLINED). If the cardholder simply closes the browser, the only status information returned to the Merchant will be the 850 Notification. Based upon the Merchant Agreement, additional notifications may also be originated by GPN DATA and sent to the Merchant; therefore, it will be necessary for the Merchant to prepare for unsolicited notifications.

The 850 Notification will also contain details of the reason for the Decline, such as "Insufficient funds" or of fraudulent activity such as "Pickup card – Stolen card"

## 850- Command Notification

Description: Notifies the merchant about the result of a submitted charge as well as most other commands except for Refunds (760). The Notification will be sent to the Notification URL supplied by the merchant.

Direction: From Gateway Server to merchant Server.

Posting URL: must be provided by merchant

Posting data (posted as urlencoded field strdata=xmldata):

## 850 Notification Sample

```

<?xml version="1.0"
encoding="utf-8" ?>
<transaction>
<apiUser></apiUser>
<apiPassword></apiPassword>
<apiCmd>850</apiCmd>
<merchanttransid></merchanttransid>
<gatetransid></gatetransid>
<amount></amount>
<curcode></curcode>
<state></state>
<merchantspecific1></merchantspecific1>
<merchantspecific2></merchantspecific2>
<merchantspecific3></merchantspecific3>
<merchantspecific4></merchantspecific4>
<merchantspecific5></merchantspecific5>
<description></description>
<time></time>
<checksum></checksum>
</transaction>

```

## 850 Notification Data description

Node	Include in Checksum (in this order)	Required	Data Type	Description
apiUser	<b>YES (1)</b>	<b>YES</b>	Character max 50.	The API user assigned by the gateway to the merchant. Can be obtained from the website inside your merchant account.
apiPassword	<b>YES (2)</b>	<b>YES</b>	Character max 50.	The API password assigned by GPN DATA to the merchant. Can be obtained from the website inside your merchant account.
apiCmd	<b>YES (3)</b>	<b>YES</b>	850	3 digit numeric code of the transaction will be 850
merchanttransid	<b>YES (4)</b>	<b>YES</b>	Character max 50.	Unique merchant transaction id received in the charge request.
gatetransid				Unique gateway internal transaction id.
amount	<b>YES (5)</b>	<b>YES</b>	Number with point decimal	Full amount to charge in US Dollars, the amount will be specified in Dollars and cents. Decimal separator must be point (.). example:125.45 = One hundred and twenty-five dollars with forty-five cents.
curcode	<b>YES (6)</b>	<b>YES</b>	ISO 3 character code	Must be USD unless other currencies are approved in the Merchant Profile. Current supported currencies are: USD, EUR and GBP

state		YES	Character	<p>Current state of the transaction.</p> <p>AUTHORIZED: Funds from Customers Credit Card account have been “reserved” but not charged to the account.</p> <p>CAPTURED: Customer's Credit Card account has been charged with the amount of the transaction.</p> <p>DECLINED: transaction was declined.</p> <p>PENDING: Transaction is in pending state</p> <p>CANCELED: authorization of the transaction was canceled</p> <p>CREDITED: Customer's Credit Card was credited with the amount of the transaction</p> <p>ERROR: An error occurred during transaction processing</p> <p>REVERSED: Transaction was reversed</p> <p>REFUNDED: Transaction was refunded</p> <p>REFUND PENDING: Refund is waiting for staff confirmation</p> <p>CHARGEBACK: means that a chargeback for the transaction was received by the gateway.</p>
merchantspecific1		YES	Character max 200.	Merchant specific data for merchant use only. Data sent is the same as was sent in the 700 transaction.
merchantspecific2		YES	Character max 200.	Merchant specific data for merchant use only. Data sent is the same as was sent in the 700 transaction.
merchantspecific3		YES	Character max 200.	Merchant specific data for merchant use only. Data sent is the same as was sent in the 700 transaction.
merchantspecific4		YES	Character max 200.	Merchant specific data for merchant use only. Data sent is the same as was sent in the 700 transaction.
merchantspecific5		YES	Character max 200.	Merchant specific data for merchant use only. Data sent is the same as was sent in the 700 transaction.
description		NO	Character max 100.	Additional descriptive information about this transaction status.
time		YES	Character 25.	Date and time of notification generation in ISO 8601 format, example: 2015-11-07T15:04:42+00:00
checksum		YES	Character 36	SHA-1 calculated using the fields specified by the transaction plus the API key assigned by GPN DATA to the merchant. This field must be calculated using the fields marked as “Include in checksum” plus the API key at the end.

## 850 Notification Response

```
<?xml version="1.0" encoding="utf-8" ?>
<transaction>
  <result></result>
  <gatetransid></gatetransid>
  <merchanttransid></merchanttransid>
  <errorcode></errorcode>
  <errormessage></errormessage>
</transaction>
```

## 850 Response Data description

Node	Required	DataType	Description
------	----------	----------	-------------

result	<b>YES</b>	Character	Result indicates the result of the transaction. OK: means transaction notification was received and accepted. ERROR: means error detected in the notification from the gateway.
gatetransid	<b>YES</b>		Unique gateway internal transaction id.
merchanttransid	<b>YES</b>	Character max 50.	Unique merchant transaction id received in the request.
errorcode	<b>NO</b>	Character 3	Optional Error Code from the Merchant
errormessage	<b>NO</b>	Character max 100	Optional description of the error.

Character max 50.