



JetPay Hosted Payment Solution (HPS)

Development and Integration Guide

<u>1 INTRODUCTION</u>	- 5 -
PROCESS OVERVIEW	- 5 -
<u>1A TYPICAL JETPAYHPS INTEGRATION FOR CREDIT CARDS</u>	- 6 -
SAMPLE OF HIDDEN POST VARIABLES:	- 6 -
SAMPLE OF GENERATED URL	- 6 -
SAMPLE OF COMPLETE IFRAME CODE	- 6 -
<u>2 MERCHANT REQUIREMENTS (CREDIT CARDS)</u>	- 7 -
<u>2A MERCHANT DATA COLLECTION (CREDIT CARDS)</u>	- 8 -
<u>2B PLANNING THE IFRAME (CREDIT CARDS)</u>	- 9 -
<u>AS STATED IN THE INTRODUCTION OF THIS DOCUMENT, THE JETPAYHPS PRODUCT IS DISPLAYED ON THE MERCHANTS SITE BY THE USE OF HTML COMPLIANT IFRAMES.</u>	- 9 -
GENERAL IFRAME INFORMATION	- 9 -
<u>THE <IFRAME> TAG DEFINES AN INLINE FRAME THAT CONTAINS ANOTHER DOCUMENT, AND IS SUPPORTED BY ALL MAJOR BROWSERS – INTERNET EXPLORER, FIREFOX, OPERA, CHROME, SAFARI</u>	- 9 -
P3P HEADERS FOR USE WITH 3 RD PARTY COOKIES.	- 9 -
STANDARD ATTRIBUTES	- 10 -
OPTIONAL ATTRIBUTES	- 10 -
SALE AND AUTHONLY WITH AVS CHECKING AND EMAIL	- 11 -
SALE AND AUTHONLY WITH EMAIL AND WITHOUT AVS CHECKING	- 12 -
SALE AND AUTHONLY WITHOUT EMAIL OR AVS CHECKING	- 13 -
TOKINIZATION ONLY	- 14 -
<u>2C IFRAME CARD FORM CSS AND ERROR CHECKING</u>	- 15 -
VISUAL AWARENESS, REQUIRED FIELD CHECKING/VALIDATION AND FORM FIELD INSTRUCTION	- 15 -
<u>3 GENERATING THE IFRAME URL (CREDIT CARDS)</u>	- 18 -

<u>3A VARIABLES FOR URL GENERATION (CREDIT CARDS)</u>	- 18 -
MERCHANT VARIABLES (CREDIT CARD)	- 18 -
MERCHANT TRANSACTION TYPE GROUPS (CREDIT CARD)	- 18 -
MERCHANT URL'S (CREDIT CARD)	- 19 -
AVS INFORMATION (BILLING ADDRESS) IF AVS CHECKING IS TURNED ON FOR THE MERCHANT ACCOUNT.	- 20 -
SPECIAL FUNCTIONS (CREDIT CARD)	- 21 -
OPTIONAL VARIABLES (CREDIT CARD)	- 21 -
<u>4 TRANSACTION RESPONSE (CREDIT CARDS)</u>	- 23 -
TRANSACTION RESPONSE VARIABLES FOR APPROVED TRANSACTIONS(CREDIT CARD)	- 23 -
VARIABLES RETURNED IF SENT BY THE MERCHANT(CREDIT CARD)	- 24 -
<u>5 JETPAYHPS FOR CHECKS (ACH)</u>	- 25 -
PROCESS OVERVIEW	- 25 -
<u>5A TYPICAL JETPAYHPS INTEGRATION (ACH)</u>	- 26 -
SAMPLE OF HIDDEN POST VARIABLES:	- 26 -
SAMPLE OF GENERATED URL	- 26 -
SAMPLE OF COMPLETE IFRAME CODE	- 26 -
<u>6 MERCHANT REQUIREMENTS (ACH)</u>	- 28 -
<u>6A PLANNING THE IFRAME (ACH)</u>	- 29 -
GENERAL IFRAME INFORMATION	- 29 -
P3P HEADERS FOR USE WITH 3 RD PARTY COOKIES.	- 29 -
STANDARD ATTRIBUTES	- 30 -
OPTIONAL ATTRIBUTES	- 30 -
SALE AND AUTHONLY WITH AVS CHECKING AND EMAIL	- 31 -
<u>6B IFRAME ACH FORM CSS AND ERROR CHECKING</u>	- 32 -
<u>7 GENERATING THE IFRAME URL (ACH)</u>	- 33 -
<u>7A VARIABLES FOR URL GENERATION (ACH)</u>	- 33 -
MERCHANT VARIABLES (ACH)	- 33 -
ACH TRANSACTION TYPE GROUP	- 33 -
ACH STANDARD ENTRY CODE CLASS (SEC CODE)	- 34 -
MERCHANT URL'S (ACH)	- 35 -

SPECIAL FUNCTIONS (ACH)	- 35 -
OPTIONAL VARIABLES (ACH)	- 36 -
<u>8 TRANSACTION RESPONSE (ACH)</u>	<u>- 37 -</u>
TRANSACTION RESPONSE VARIABLES FOR APPROVED TRANSACTIONS (ACH)	- 37 -
VARIABLES RETURNED IF SENT BY THE MERCHANT (ACH)	- 37 -
<u>9 CERTIFICATION TESTING</u>	<u>- 38 -</u>
<u>9A STANDARD TRANSACTION PROCESSING TEST SUITE</u>	<u>- 38 -</u>
<u>9B CVV2 TEST CASES</u>	<u>- 38 -</u>
<u>9C ADDRESS VERIFICATION TEST CASES</u>	<u>- 39 -</u>
<u>9D CHECKING ACCOUNT AND SAVINGS ACCOUNT TRANSACTIONS (ACH)</u>	<u>- 40 -</u>
<u>APPENDIX</u>	<u>- 41 -</u>
VISA AVS RESULT CODES(CREDIT CARD)	- 41 -
MASTER CARD AVS RESULT CODES (CREDIT CARD)	- 42 -
URL GENERATOR CREDIT CARD (SAMPLE IN PHP)	- 43 -
URL GENERATOR FOR ACH (SAMPLE IN PHP)	- 47 -
SAMPLE HTML FOR SETTING <IFRAME> TAG	- 50 -
COUNTRY LIST	- 51 -

1 Introduction

The JetPay Hosted Payment System or HPS was developed as a tool to assist merchants that either can not or do not have the technical means to meet the rigors of PCI-DSS Compliance for collecting and transmitting credit card data. JetPay HPS consists of a simple merchant control script that collects POST data, and generates an HTML compliant URL. This URL is then used by the merchant as the source link for an imbedded iFrame with in their site, which loads a credit card data collection form that is hosted on JetPay's PCI compliant servers.

Process Overview

The diagram below provides an over view of the general flow of the JetPay HPS product.



1. Merchant Checkout Page - This is generally the customer review page showing the order and the total amount to be paid. This page will POST the open filed variables to the JetPayHPS Merchant Script. This includes the amount to charge, the return URL's, Customer ID information to track the transaction.
2. JetPayHPS Merchant Script - The merchant script is provided to the merchant by JetPay and is hosted on the merchants' server like any other page in their site. The script collects the variable data from the merchant POST and combines it with specific merchant information to generate the URL needed to load the JetPayHPS payment form in the merchant's iFrame.
3. Merchant's iFrame Page - The merchants page that contains the iFrame where JetPayHPS loads based on the URL generated in the JetPayHPS Merchant Script.
4. JetPay Payment Engine - Once the customer has completed the JetLink card payment form, the data is submitted to the JetPay Payment Engine for processing.
5. Approved or Decline - After the transaction has been processed, JetPayHPS return the user back to the merchants site via the approved or decline URL provided by the merchant. Also, the APPROVED or DECLINED transaction data is posted to the merchant for collection.
6. Merchant Control - From the APPROVED or DECLINED return link, the customer is returned to the full control of the merchant's site.

1a Typical JetPayHPS Integration for Credit Cards

A typical integration of the JetPayHPS product is as follows. The customer at the merchant site has completed adding items to their shopping cart. The customer is moved to the cart page where they are provided a list of what items are in their cart, the amount and quantities, tax, shipping and a total amount for the purchase. At this point the customer is prompted to click a "Pay Now" button. Attached to the button in the <form> tag of this page there are a number of "hidden" variables for the POST method of the form.

Sample of hidden POST variables:

```
<input type="hidden" name="name" value="Bob Mallard" />
<input type="hidden" name="address1" value="3361 Boyington Dr" />
<input type="hidden" name="address2" value="Suite 180" />
<input type="hidden" name="city" value="Carrollton" />
<input type="hidden" name="state" value="Texas" />
<input type="hidden" name="zip" value="75006" />
<input type="hidden" name="country" value="USA" />
<input type="hidden" name="amount" value="165.00" />
<input type="hidden" name="email" value="bob@jetpay.com" />
<input type="hidden" name="orderNum" value="17588933" />
<input type="hidden" name="ud1" value="Platinum Customer" />
<input type="hidden" name="ud2" value="Coupon 10DISC" />
<input type="hidden" name="ud3" value="SHIP GRND" />
```

Once the "Pay Now" Button is clicked this data is sent as a POST to the script that generates the URL to call the JetPayHPS product.

Sample of Generated URL

```
https://jetpayhps_cardform_location.php?cid=ec67c493eb196701be4992f994085ad&tid=TESTTERMINAL&transType=SALE&verSub=JL4.10SALE&amount=165.00&retUrl=https%3A%2F%2Flocalhost%2Fjetlinkhps%2Fpost_transaction%2Fcc_approve_complete.php&decUrl=http%3A%2F%2Flocalhost%2Fjetlinkhps%2Fpost_transaction%2Fcc_decline_complete.php&cancelUrl=http%3A%2F%2Fwww.jetpay.com&name=Bob%20Mallard&css=https%3A%2F%2Flocalhost%2Fjetlinkhps%2Fcss%2Fjetlink_cc.css&email=bob%40jetpay.com&getAvs=true&orderNum=17588933&ud1=Platinum%20Customer&ud2=Coupon%2010DISC&ud3=SHIP%20GRND&urlEnd=END
```

From here the generated URL is inserted into the <iframe> tag as the value of "src"

Sample of complete iFrame code

```
<iframe src="
jetpayhps_cardform_location.php?cid=ec67c493eb196701be4992f994085ad&tid=TESTTERMINAL&transType=SALE&verSub=JL4.10SALE&amount=165.00&retUrl=merchants_approved_return_url&decUrl=the_merchants_decline_return_url&cancelUrl=the_merchants_cancel_url&name=Bob%20Mallard&email=bob%40jetpay.com&getAvs=true&orderNum=17588933&ud1=Platinum%20Customer&ud2=Coupon%2010DISC&ud3=SHIP%20GRND&urlEnd=END"
width="550px" height="800px" frameBorder="0"></iframe>
```

Once the URL is inserted in the "src" of the <iframe> tag code the iFrame will load the card form from the JetPay PCI Compliant servers. After the card form loads, any information that can be pre populated in the card form that has been passed in the URL will be set. Using the above URL these properties would be set in the card form.

Customer Name.
Customer email Address.
Customer Billing Address.
Total Amount to be paid.

The customer can make changes to any of these fields at this point as they are prompted to verify the information provided. The customer will also enter their credit card number, expiration date and security code.

Once the customer is satisfied with the information that has been entered, they will click the "Submit" button.

The card information is collected along with the posted data for AVS Checking and is submitted to JetPay's front end for processing. After the transaction has taken place the JetPayHPS product will read the return response and send the customer to either the APPROVE or DECLINED URL as provided by the merchant along with data about the transaction. This will conclude JetPay's connection to the merchant for this transaction.

For a list of the transaction variables that are supplied back to the merchant at the conclusion of a transaction please see section 4 Transaction Response.

After the customer is returned to the merchant, the transaction data can also be accessed through JetPay's GetReporting application.

2 Merchant Requirements (Credit Cards)

While the JetPayHPS product has been developed to make integration to a merchant's site as easy as possible there are a few requirements that every merchant must meet in order to use the product.

1. Merchant SSL Certificate – Each merchant should have their own signed SSL Certificate.
2. Page customization and custom scripting – Merchant's must be able to customize their own pages and run custom scripts within their web site. If the merchant is using a template driven host site, they will need to work with their host to complete the customization required.
3. Receive POST data under HTTPS from the JetPayHPS product – At the conclusion of a transaction the merchant's site must be able to receive HTTPS POST data from the JetPayHPS product as record of the transaction OR they must be subscribed to JetPay's GetReporting service to track their daily transactions.

2a Merchant Data Collection (Credit Cards)

In order for the JetPayHPS to process a transaction there are certain data pieces that will need to be either collected from the customer or provided by the merchant prior to generating the iFrame URL. These data pieces included the following.

Required Data

1. **Customer ID** – This is an identifier used by the merchant that will be passed back with the transaction data so the merchant can tie the customer and the transaction together.
2. **Total Amount** – This is the TOTAL AMOUNT that is to be paid. This amount must include all taxes, shipping, fees etc as there is no calculation done on the payment form.

Optional Data

1. **Customer Billing Address** – If the merchant has this information or will collect this information in another part of their site they may provide as part of the standard data collection.
2. **Order Number** – The order number is alphanumeric free forms filed that can be used by the merchant to further identify the transaction and customer.
3. **Free Form Field 1, 2, 3** – JetPay provides for up to three (3) free form fields that may be used by the merchant to further identify the transaction, customer or other important information about the transaction or customer as they need.
4. **Email Address** – The primary or email address on file for the customer.

The above fields are described in more detail in section “3a Variables for URL Generation” of this document.

2b Planning the iFrame (Credit Cards)

As stated in the Introduction of this document, the JetPayHPS product is displayed on the merchants site by the use of HTML compliant iFrames.

General iFrame Information

The <iframe> tag defines an inline frame that contains another document, and is supported by all major browsers – Internet Explorer, Firefox, Opera, Chrome, Safari

P3P Headers for use with 3rd party cookies.

Since the main communication method being used to contact the merchant domain to the JetPayHPS domain is an iFrame some special considerations need to be taken into account. These considerations revolve around the use of 3rd party cookies and the issues that this can cause. It is recommended that in the HEAD of the merchant page that contains the iFrame you insert a compact privacy policy header that matches the policy header used in the JetPayHPS product.

Below is the JetPayHPS Compact Privacy Headers in various code sets. Insert the P3P Header between your <head></head> tags in your html.

ASP
<pre>HttpContext.Current.Response.AddHeader("p3p", "CP="CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR\"")</pre>
PHP
<pre>header('P3P: CP="CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR"');</pre>
JSP
<pre>response.setHeader("P3P","CP='CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR'")</pre>
ColdFusion
<pre><cfheader name="P3P" value="CP="CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR'" /></pre>

Standard Attributes

The <iframe> tag supports the following standard attributes:

Attribute	Value	Description	DTD
class	<i>classname</i>	Specifies a classname for an element	TF
id	<i>id</i>	Specifies a unique id for an element	TF
style	<i>style_definition</i>	Specifies an inline style for an element	TF
title	<i>text</i>	Specifies extra information about an element	TF

Optional Attributes

DTD indicates in which [HTML 4.01/XHTML 1.0 DTD](#) the attribute is allowed. S=Strict, T=Transitional, and F=Frameset.

Attribute	Value	Description	DTD
align	left right top middle bottom	Deprecated. Use styles instead. Specifies the alignment of an iframe according to surrounding elements	TF
frameborder	1 0	Specifies whether or not to display a border around an iframe	TF
height	<i>pixels</i> %	Specifies the height of an iframe	TF
longdesc	<i>URL</i>	Specifies a page that contains a long description of the content of an iframe	TF
marginheight	<i>pixels</i>	Specifies the top and bottom margins of an iframe	TF
marginwidth	<i>pixels</i>	Specifies the left and right margins of an iframe	TF
name	<i>name</i>	Specifies the name of an iframe	TF
scrolling	yes no auto	Specifies whether or not to display scrollbars in an iframe	TF
src	<i>URL</i>	Specifies the URL of the document to show in an iframe	TF
width	<i>pixels</i> %	Specifies the width of an iframe	TF

The iFrame can be loaded between the HEADER and FOOTER of the merchants page so that the merchants logo etc to make the page look as close as possible to the merchants site.

Depending on your implementation of the JetPayHPS product the height of the iFrame will vary.

SALE and AUTHONLY with AVS Checking and email

Attribute	Value
width	550px
height	800px

The image shows a payment form with the following fields and elements:

- Card Holder Name ***: Text input field.
- Card Number ***: Text input field.
- Expiration Date**: Two dropdown menus for month (mo.) and year (yr.).
- Security Code ***: Text input field.
- Total Amount**: **1.00**
- Email Address ***: Text input field.
- Billing Address**: Text input field.
- Street Address***: Text input field.
- Address Line 2 (optional)**: Text input field.
- City***: Text input field.
- State / Province / Region***: Text input field.
- Postal / Zip Code***: Text input field.
- Country***: Dropdown menu with "United States of America" selected.
- Submit**: Button.
- Payments Processed by JetPay, LLC**: Text.
- PCI DSS Certified Class 1 Service Provider**: Logo.

Dimensions: 550PX width, 800PX height.

Example iFrame Tag

```
<iframe src="generated_jetpayhps_url_goes_here" width="550px" height="800px"
frameborder="0" scrolling="no"></iframe>
```

SALE and AUTHONLY with email and without AVS Checking

Attribute	Value
width	550px
height	560px

The image shows a payment form with the following fields and elements:

- Card Holder Name * (text input)
- Card Number * (text input)
- Expiration Date (dropdown for month, text for 'mo.', dropdown for year, text for 'yr.')
- Security Code * (text input)
- Total Amount: **1.00**
- Email Address * (text input)
- Submit button
- Payments Processed by JetPay, LLC
- PCI DSS Certified Class 1 Service Provider logo

Dimensions: 550px width, 560px height.

Example iFrame Tag

```
<iframe src="generated_jetpayhps_url_goes_here" width="550px" height="560px"
frameborder="0" scrolling="no"></iframe>
```

SALE and AUTHONLY without email or AVS Checking

Attribute	Value
width	550px
height	500px

The image shows a payment form with the following fields and elements:

- Card Holder Name * (text input)
- Card Number * (text input)
- Expiration Date (dropdown for month, "mo.", dropdown for year, "yr.")
- Security Code * (text input)
- Total Amount: **1.00**
- Submit button
- Payments Processed by JetPay, LLC
- PCI DSS Certified Class 1 Service Provider logo

Dimensions: 550px width and 500px height.

Example iFrame Tag

```
<iframe src="generated_jetpayhps_url_goes_here" width="550px" height="500px"
frameborder="0" scrolling="no"></iframe>
```

TOKINIZATION ONLY

Attribute	Value
width	550px
height	420px

The image shows a payment form with the following fields and elements:

- Card Holder Name * (text input)
- Card Number * (text input)
- Expiration Date (dropdown for month, text input for month, dropdown for year, text input for year)
- Security Code * (text input)
- Submit button
- Text: Payments Processed by JetPay, LLC
- Logo: PCI DSS Certified Class 1 Service Provider

Dimensions: 550px width, 420px height.

Example iFrame Tag

```
<iframe src="generated_jetpayhps_url_goes_here" width="550px" height="420px"
frameborder="0" scrolling="no"></iframe>
```

2c iFrame Card Form CSS and Error Checking

The JetPayHPS product uses CSS Styling and scripting to set the fields, layout as well as form field validation. Below is some general information about the use of CSS within the product.

Visual Awareness, Required Field Checking/Validation and Form Field Instruction

Through the use of CSS the card form shows the current focus as well as helpful instructions on each of the required fields.

Each field has 4 states that it can / will display

Normal

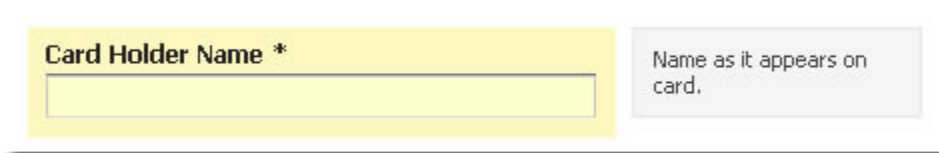
This is the default state of the form fields.



The screenshot shows a form field with the label "Card Holder Name *" above it. The input field is empty and has a thin border. The entire form area is enclosed in a thin black border.

On Focus

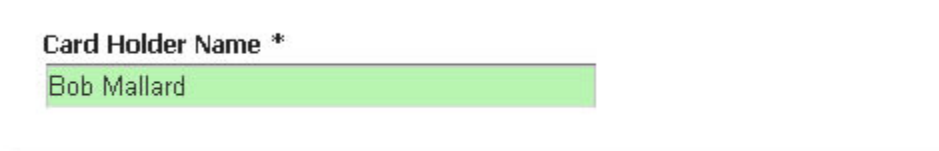
This state is set when the customer clicks inside a required field. Note the field is highlighted to let the customer know where they are on the form and a field "hint" box is displayed when the focus is set.



The screenshot shows the same form field as in the normal state, but now the input field and its label are highlighted with a yellow background. To the right of the input field, a grey box contains the text "Name as it appears on card." The entire form area is enclosed in a thin black border.

Data Entered – Field Complete

When the field has been completed and the information validated by pattern or field type matching the form field color will alert that the field is correct.



The screenshot shows the same form field as in the normal state, but now the input field contains the text "Bob Mallard" and is highlighted with a green background. The entire form area is enclosed in a thin black border.

Skipped Required Field.

If the customer misses filling in one of the required form field the user is alerted by a background color change of the field.

Card Holder Name *

These attributes to the fields, aid in not only collecting the required data fields but also act as markers for those customers that may be visually impaired. Further, if a customer skips a field and has not put focus on the field to activate the checking, when they click the submit button the form checks the required fields for completeness. If the form finds that a required field that has no data or if the data is incorrect based on the patterns that are supposed to be present. The form will not be allowed to be submitted until the errors or missing data is corrected.

Card Validation Check

The JetPayHPS also performs a Luhn10s pre check to validate the cards that are entered into the Card Number field. This check helps to ensure that the cards being used are valid card numbers and aids in cutting down the number of declines that may be the result of an incorrectly entered card number.

Card Number Field Activated

Card Number *

Enter the card number do not include dashes "-" or spaces " ",

Note the "hint" box providing direction for the field.

Invalid Card Number Entered

Card Number *



Enter the card number do not include dashes "-" or spaces " ",

Valid Card Number Entered

<p>Card Number *</p> <input type="text" value="4000300020001000"/>		<p>Enter the card number do not include dashes "-" or spaces " " ,</p>
---------------------------------------------------------------------------	-----------------------------------------------------------------------------------	------------------------------------------------------------------------

When a "valid" card number is entered, the logo of the card type is displayed. Currently, the JetPayHPS product supports the validation of the following card types:



While the form can accept and perform checks to validate the card types listed above, not all card types are available through your merchant account. You must have a Merchant Account with each of the card associations in order to take a particular card. Please contact your JetPay Merchant Account Representative for information on how to obtain a merchant account for a particular card association.

3 Generating the iFrame URL (Credit Cards)

After you have completed the planning for the iFrame and the page has been constructed you will now focus on the generation of the iFrame URL.

3a Variables for URL Generation (Credit Cards)

The following table lists the variable names and requirements for each.

For a sample of a complete URL generation script see URL Generation Credit Cards in the Appendix section.

Merchant Variables (Credit Card)

Variable	Value	Description	Required
\$cid	Merchant Defined Alphanumeric Max Length = 28 characters	This variable is a "Customer Id". The merchant must provide some sort of identifier for the customer that can be used to link the customer and the transaction when the transaction data is returned to the merchant.	Yes
\$tid	TESTTERMINAL Must be provided exactly as provided by Jetpay.	This is the Terminal Id provided to the merchant by JetPay. Please note that only one terminal Id may be used per instance of the JetPayHPS product. If you have more than one site you will need a separate terminal Id for each site.	Yes
\$amount	Amount must be in USD and formatted as DDDD.CC	This is the total amount that is to be charged to the customer. This amount MUST include all taxes, fees, and other adjustments. No calculations will be done within the JetPayHPS product	Yes
\$name	Max Length = 50 Characters.	First and Last Name of the customer. ***Note: Some merchants may use separate fields for First and Last Names of the customer. In this instance allowance has been made to use the following: \$fName Max Length = 20 \$lName Max Length = 20	Yes

Merchant Transaction Type Groups (Credit Card)

You may only use one Transaction Type Group at a time.*

Transaction Type Groups – AUTHONLY

\$transType	AUTHONLY		Yes
\$verSub	JL4.10AUTH		Yes

Transaction Type Groups – AUTHONLY w/Tokenization

\$transType	AUTHONLY		Yes
\$verSub	JL4.10AOTO		Yes
\$tokenTrans	true		Yes
Transaction Type Groups – SALE			
\$transType	SALE		Yes
\$verSub	JL4.10SALE		Yes
Transaction Type Groups – SALE w/Tokenization			
\$transType	SALE		Yes
\$verSub	JL4.10SATO		Yes
\$tokenTrans	true		Yes
Transaction Type Groups – TOKENIZATION Only			
\$transType	TOKENIZE		Yes
\$verSub	JL4.10TOKEN		Yes

Your Transaction Type will be determined when the merchant account is created. Attempts to use multiple or incorrect Transaction Types will result in declined transaction and possible suspension of the merchant account.

******SPECIAL NOTE******

When using the \$transType “TOKENIZE” you do not need to supply AVS information as it will not be checked. You may send AVS when using a TOKEN to complete a transaction such as a SALE using a TOKEN in place of the card information. Also, in a “TOKENIZE” transaction you will not receive a response for Address Match, Zip Match or CVV Match as they are not checked for TOKENIZE transactions.

Merchant URL's (Credit Card)

Variable	Value	Description	Required
\$retUrl	<i>Merchant Defined</i>	The link to the page within the merchant site where the customer will be returned to if the transaction is APPROVED. This page will also serve as the collection point for the transaction data sent back to the merchant from the JetPayHPS product.	Yes
\$decUrl	<i>Merchant Defined</i>	This link is to a page within the merchant's site where the customer will be returned to if the transaction is DECLINED. This page will also collect transaction data from the JetPayHPS Product	YES
\$cancelUrl	<i>Merchant Defined</i>	This link is to a page within the merchants site where the customer will be returned to if the customer decides to CANCEL the purchase. No transaction data will be returned.	No

AVS Information (Billing Address) if AVS Checking is Turned on for the Merchant Account.

Variable	Value	Description	Required
\$address1		This is the first line of the billing street address of the customer.	Yes
\$address2		This is the second line of the billing street address of the customer. Generally this is an Apartment number etc.	No
\$city		The City of the billing address of the customer	Yes
\$state	IF is a State it may be the two (2) letter abbreviation or the full name of the state. Example: TX or TEXAS ELSE the full name is required. Max Length = 25	The State/Province/Region of the billing address of the customer.	Yes
\$zip		This is the Zip or Postal Code of the billing address	Yes
\$country	The country must be provided in the ISO 3166 Standard three (3) Character Codes. Example: United States of America = USA* ***Note: If no country is sent the default will be USA.*** Max Length = 3	The country of the billing address	No

***A list of the three (3) characters to full name of countries is located in the Appendix under Country List.**

Special Functions (Credit Card)

Variable	Value	Description	Required
\$getAvs	true	<p>If the merchant is subscribing to JetPay AVS Checking but does not wish to collect the Billing Address prior to the calling the JetPayHPS product. This will invoke the Billing Address fields to become active on the card collection form.</p> <p>***Note: If the merchant IS collecting BILLING Information and passing this information to the URL, the card form page of the JetPayHPS will auto fill and display the Billing Address fields automatically. In this instance the variable does not need to be set.***</p> <p>***Note: If you are using the tranType of "TOKENIZE", NO AVS information needs to be sent. IF AVS is sent it will not be collected or presented back to the customer***</p>	Yes
\$getEmail	true	<p>If the merchant would like for the JetPayHPS to collect the customers email address enable this variable.</p> <p>***Note: Merchants would choose this option if they are going to send an email receipt to the customer after the transaction has been completed and the customer has returned to the merchant's web site. Further, if the merchant has the email and is sending it as part of the URL build this variable does not need to be set.</p>	No
\$urlEnd	end	This is the last variable that is to be included in the URL generation. This variable tells the JetPayHPS cardform that there is no further data to parse.	Yes

Optional Variables (Credit Card)

Variable	Value	Description	Required
\$orderNum	Merchant Defined Max Length = 12	Open Alphanumeric field it CAN NOT contain special characters.	No
\$ud1	Merchant Defined Max Length = 24	Open Alphanumeric field it CAN NOT contain special characters.	No
\$ud2	Merchant Defined Max Length = 24	Open Alphanumeric field it CAN NOT contain special characters.	No

\$ud3	Merchant Defined Max Length = 24	Open Alphanumeric field it CAN NOT contain special characters.	No
-------	-------------------------------------	----------------------------------------------------------------	----

4 Transaction Response (Credit Cards)

The following table list the variable names and requirements for each.

Transaction Response Variables for APPROVED Transactions (Credit Card)

Variable	Value	Description	Returned
\$cid		Returned of received data from merchant	Yes
\$transId		18 Character alphanumeric string as id for the transaction	Yes
\$obfCcNum	XXXXXX*****XXXX	The first 6 and last 4 of the card number	Yes
\$expDate	MMYY	Month and Year of Expiration Date	Yes
\$approval		6 Character alphanumeric string sent by card association as their transaction identifier.	Yes
\$actionCode	000 – 999	3 Digit code to indicate the APPROVAL or DECLINE Code 000 = APPROVED 001 – 999 = DECLINE A full listing of decline codes in .xls format can be obtained by sending an email to assist@jetpay.com and request a listing of Decline Codes.	Yes
\$name \$fName \$lName		Name of customer returned in the format sent by the merchant	Yes
\$addressMatch	*SEE AVS RESPONSES*	If AVS Checking is turned on in the merchants account a response will be sent when checking the address.	*
\$zipMatch	*SEE AVS RESPONSES*	If AVS Checking is turned on in the merchants account a response will be sent when checking the zip. (US ONLY)	*
\$cvvMatch	M = Match N = No Match P = Not Processed S = Missing on Card U = Issuer Not Certified	If CVV Checking is turned on in the merchants account a response will be sent.	*
\$token		Returned if \$transType was set to SALE w/ Tokenization AUTHONLY w/ Tokenization or TOKENIZE	^

*Will be returned if sent by the merchant or if collected by the JetPayHPS card form other wise will not be sent.

^ Retuned based on \$transType variable.

Variables Returned if sent by the Merchant (Credit Card)

Variable	Value
\$address1	As sent by the merchant or as collected by card form
\$address2	As sent by the merchant or as collected by card form
\$city	As sent by the merchant or as collected by card form
\$state	As sent by the merchant or as collected by card form
\$zip	As sent by the merchant or as collected by card form
\$country	As sent by the merchant or as collected by card form
\$orderNum	As sent by the merchant
\$ud1	As sent by the merchant
\$ud2	As sent by the merchant
\$ud3	As sent by the merchant

5 JetPayHPS for Checks (ACH)

Through the JetPayHPS product you may also be able to accept Checks on line in the form of ACH payments. While the methods for accepting ACH are very similar to accepting Credit Cards there are some distinct differences. For information about being able to accept checks online please contact your Jetpay merchant account representative BEFORE beginning development for taking checks online. **Not all merchants will qualify to take checks.**

Process Overview

The diagram below provides an over view of the general flow of the JetPay HPS product.



7. Merchant Checkout Page - This is generally the customer review page showing the order and the total amount to be paid. This page will POST the open filed variables to the JetPayHPS Merchant Script. This includes the amount to charge, the return URL's, Customer ID information to track the transaction.
8. JetPayHPS Merchant Script - The merchant script is provided to the merchant by JetPay and is hosted on the merchants' server like any other page in their site. The script collects the variable data from the merchant POST and combines it with specific merchant information to generate the URL needed to load the JetPayHPS payment form in the merchant's iFrame.
9. Merchant's iFrame Page - The merchants page that contains the iFrame where JetPayHPS loads based on the URL generated in the JetPayHPS Merchant Script.
10. JetPay Payment Engine - Once the customer has completed the JetLink card payment form, the data is submitted to the JetPay Payment Engine for processing.
11. Approved or Decline - After the transaction has been processed, JetPayHPS return the user back to the merchants site via the approved or decline URL provided by the merchant. Also, the APPROVED or DECLINED transaction data is posted to the merchant for collection.
12. Merchant Control - From the APPROVED or DECLINED return link, the customer is returned to the full control of the merchant's site.

5a Typical JetPayHPS Integration (ACH)

A typical integration of the JetPayHPS product is as follows. The customer at the merchant site has completed adding items to their shopping cart. The customer is moved to the cart page where they are provided a list of what items are in their cart, the amount and quantities, tax, shipping and a total amount for the purchase. At this point the customer is prompted to just pay with a credit card, a button that indicates they would like to pay by check. Attached to this button is a hidden form just as with credit cards but this form contains a little less information than with the credit cards.

Sample of hidden POST variables:

```
<input type="hidden" name="name" value="Bob Mallard" />
<input type="hidden" name="email" value="bob@jetpay.com" />
<input type="hidden" name="amount" value="165.00" />
<input type="hidden" name="orderNum" value="17588933" />
<input type="hidden" name="ud1" value="Platinum Customer" />
<input type="hidden" name="ud2" value="Coupon 10DISC" />
<input type="hidden" name="ud3" value="SHIP GRND" />
```

Since there is no address verification associated with ACH these values are removed.

Once the "Pay by Check" Button is clicked this data is sent as a POST to the script that generates the URL to call the JetPayHPS product.

Sample of Generated URL

```
https://jetpayhps_checkform_location.php?
cid=ab4ed0f1f47ce2db92b4108ff9fbc2cc&tid=TESTTERMINAL&transType=CHECK&sec=WEB&ve
rSub=iFrame%202.1.1%20ACH&amount=1.00&retUrl=https
%3A%2F%merchant_approved_return_page.php&decUrl=
https%3A%2F%erchant_decline_return_page.php
&cancelUrl=http%3A%2F%2cancel_page.php&email=bob%40jetpay.com&name=Bob%20Mallard
&orderNum=321321321&ud1=TERMS%20AGRRED&ud2=1%25%20FEE%20for%20Check&urlEnd=END
```

From here the generated URL is inserted into the <iframe> tag as the value of "src"

Sample of complete iFrame code

```
<iframe src=" https://jetpayhps_checkform_location.php?
cid=ab4ed0f1f47ce2db92b4108ff9fbc2cc&tid=TESTTERMINAL&transType=CHECK&sec=WEB&ve
rSub=iFrame%202.1.1%20ACH&amount=1.00&retUrl=https
%3A%2F%merchant_approved_return_page.php&decUrl=
https%3A%2F%erchant_decline_return_page.php
&cancelUrl=http%3A%2F%2cancel_page.php&email=bob%40jetpay.com&name=Bob%20Mallard
&orderNum=321321321&ud1=TERMS%20AGRRED&ud2=1%25%20FEE%20for%20Check&urlEnd=EN"
width="550px" height="800px" frameborder="0"></iframe>
```

Once the URL is inserted in the "src" of the <iframe> tag code the iFrame will load the check form from the JetPay NACHA Compliant systems. After the check form loads, any information that can be pre populated in the card form that has been passed in the URL will be set. Using the above URL these properties would be set in the card form.

Customer Name.
Customer email Address.

The customer can make changes to any of these fields at this point as they are prompted to verify the information provided. The customer will also enter their credit card number, expiration date and security code.

Once the customer is satisfied with the information that has been entered, they will click the "Submit" button.

The check information is collected along with the posted data for and is submitted to JetPay's front end for processing. After the transaction has taken place the JetPayHPS product will read the return response and send the customer to either the APPROVE or DECLINED URL as provided by the merchant along with data about the transaction. This will conclude JetPay's connection to the merchant for this transaction.

For a list of the transaction variables that are supplied back to the merchant at the conclusion of a transaction please see section 4a ACH Transaction Response.

After the customer is returned to the merchant, the transaction data can also be accessed through JetPay's GetReporting application.

6 Merchant Requirements (ACH)

While the JetPayHPS product has been developed to make integration to a merchant's site as easy as possible there are a few requirements that every merchant must meet in order to use the product.

4. Merchant SSL Certificate – Each merchant must have their own signed SSL Certificate. A merchant may not use a shared certificate issued by their host.
5. Page customization and custom scripting – Merchant's must be able to customize their own pages and run custom scripts within their web site. If the merchant is using a template driven host site, they will need to work with their host to complete the customization required.
6. Receive POST data under HTTPS from the JetPayHPS product – At the conclusion of a transaction the merchant's site must be able to receive HTTPS POST data from the JetPayHPS product as record of the transaction OR they must be subscribed to JetPay's GetReporting service to track their daily transactions.

6a Planning the iFrame (ACH)

As stated in the Introduction of this document, the JetPayHPS product is displayed on the merchant's site by the use of HTML compliant iFrames.

General iFrame Information

The <iframe> tag defines an inline frame that contains another document, and is supported by all major browsers – Internet Explorer, Firefox, Opera, Chrome, Safari

P3P Headers for use with 3rd party cookies.

Since the main communication method being used to contact the merchant domain to the JetPayHPS domain is an iFrame some special considerations need to be taken into account. These considerations revolve around the use of 3rd party cookies and the issues that this can cause. It is recommended that in the HEAD of the merchant page that contains the iFrame you insert a compact privacy policy header that matches the policy header used in the JetPayHPS product.

Below is the JetPayHPS Compact Privacy Headers in various code sets. Insert the P3P Header between your <head></head> tags in your html.

ASP
<pre>HttpContext.Current.Response.AddHeader("p3p", "CP="CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR\"")</pre>
PHP
<pre>header('P3P: CP="CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR"');</pre>
JSP
<pre>response.setHeader("P3P","CP='CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR'")</pre>
ColdFusion
<pre><cfheader name="P3P" value="CP="CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR'" /></pre>

Standard Attributes

The <iframe> tag supports the following standard attributes:

Attribute	Value	Description	DTD
class	<i>classname</i>	Specifies a classname for an element	TF
id	<i>id</i>	Specifies a unique id for an element	TF
style	<i>style_definition</i>	Specifies an inline style for an element	TF
title	<i>text</i>	Specifies extra information about an element	TF

Optional Attributes

DTD indicates in which [HTML 4.01/XHTML 1.0 DTD](#) the attribute is allowed. S=Strict, T=Transitional, and F=Frameset.

Attribute	Value	Description	DTD
align	left right top middle bottom	Deprecated. Use styles instead. Specifies the alignment of an iframe according to surrounding elements	TF
frameborder	1 0	Specifies whether or not to display a border around an iframe	TF
height	<i>pixels</i> %	Specifies the height of an iframe	TF
longdesc	<i>URL</i>	Specifies a page that contains a long description of the content of an iframe	TF
marginheight	<i>pixels</i>	Specifies the top and bottom margins of an iframe	TF
marginwidth	<i>pixels</i>	Specifies the left and right margins of an iframe	TF
name	<i>name</i>	Specifies the name of an iframe	TF
scrolling	yes no auto	Specifies whether or not to display scrollbars in an iframe	TF
src	<i>URL</i>	Specifies the URL of the document to show in an iframe	TF
width	<i>pixels</i> %	Specifies the width of an iframe	TF

The iFrame can be loaded between the HEADER and FOOTER of the merchants page so that the merchants logo etc to make the page look as close as possible to the merchants site.

Depending on your implementation of the JetPayHPS product the height of the iFrame will vary.

SALE and AUTHONLY with AVS Checking and email

Attribute	Value
width	550px
height	640px

The image shows a screenshot of an ACH payment form. The form is rectangular and contains the following elements from top to bottom: a section header 'Account Information' with a note 'All fields are required'; a text input field for 'Account Holder Name *'; a text input field for 'Email Address *'; a dropdown menu for 'Account Type *'; a text input field for 'Routing Number *'; a text input field for 'Account Number *'; a text input field for 'Check Number *'; a section header 'Total Amount'; a 'Submit' button; the text 'Payments Processed by JetPay, LLC'; and the NACHA logo with the text 'NACHA Certified Compliant'. Dimension lines indicate the form's width is 550px and its height is 640px.

Unlink the iFrame generation for credit cards the iFrame for ACH is a constant size. The ACH form will not grow or shrink based on the values provided.

6b iFrame ACH Form CSS and Error Checking

CSS and error checking for ACH is the same method used for the credit card form, however, there is NO checking done for the correctness of Routing or Account numbers since these may vary in length and format from bank to bank.

7 Generating the iFrame URL (ACH)

URL generation for ACH is handled in much the same way as the URL generation for credit cards.

7a Variables for URL Generation (ACH)

The following table lists the variable names and requirements for each.

For a sample of a complete URL generation script see URL Generation ACH in the Appendix section.

Merchant Variables (ACH)

Variable	Value	Description	Required
\$cid	Merchant Defined Alphanumeric Max Length = 28 characters	This variable is a "Customer Id". The merchant must provide some sort of identifier for the customer that can be used to link the customer and the transaction when the transaction data is returned to the merchant.	Yes
\$tid	TESTTERMINAL Must be provided exactly as provided by Jetpay.	This is the Terminal Id provided to the merchant by JetPay. Please note that only one terminal Id may be used per instance of the JetPayHPS product. If you have more than one site you will need a separate terminal Id for each site.	Yes
\$amount	Amount must be in USD and formatted as DDDD.CC	This is the total amount that is to be charged to the customer. This amount MUST include all taxes, fees, and other adjustments. No calculations will be done within the JetPayHPS product	Yes
\$name	Max Length = 50 Characters.	First and Last Name of the customer. ***Note: Some merchants may use separate fields for First and Last Names of the customer. In this instance allowance has been made to use the following: \$fName Max Length = 20 \$lName Max Length = 20	No

ACH Transaction Type Group

Variable	Value	Description	Required
\$transType	CHECK	Transaction type marker for web based ACH	Yes
\$verSub	JL4.10ACH	Version subscription information	Yes

ACH Standard Entry Code Class (SEC Code)

Variable	Value	Description	Required
\$sec	WEB	Internet- Initiated Entries	Yes

Note that there are other SEC Code types that may be used based on the merchant type and activity. However, for most all internet based ACH Transaction the use of the SEC Code type of "WEB" is standard.

Merchant URL's (ACH)

Variable	Value	Description	Required
\$retUrl	<i>Merchant Defined</i>	The link to the page within the merchant site where the customer will be returned to if the transaction is APPROVED. This page will also serve as the collection point for the transaction data sent back to the merchant from the JetPayHPS product.	Yes
\$decUrl	<i>Merchant Defined</i>	This link is to a page within the merchant's site where the customer will be returned to if the transaction is DECLINED. This page will also collect transaction data from the JetPayHPS Product	YES
\$cancelUrl	<i>Merchant Defined</i>	This link is to a page within the merchants site where the customer will be returned to if the customer decides to CANCEL the purchase. No transaction data will be returned.	No

Special Functions (ACH)

Variable	Value	Description	Required
\$getEmail	true	If the merchant would like for the JetPayHPS to collect the customers email address enable this variable. ***Note: Merchants would choose this option if they are going to send an email receipt to the customer after the transaction has been completed and the customer has returned to the merchant's web site. Further, if the merchant has the email and is sending it as part of the URL build this variable does not need to be set.***	No
\$urlEnd	END	This is the last variable that is to be included in the URL generation. This variable tells the JetPayHPS cardform that there is no further data to parse.	Yes

Optional Variables (ACH)

Variable	Value	Description	Required
\$orderNum	Merchant Defined Max Length = 12	Open Alphanumeric field it CAN NOT contain special characters.	No
\$ud1	Merchant Defined Max Length = 24	Open Alphanumeric field it CAN NOT contain special characters.	No
\$ud2	Merchant Defined Max Length = 24	Open Alphanumeric field it CAN NOT contain special characters.	No
\$ud3	Merchant Defined Max Length = 24	Open Alphanumeric field it CAN NOT contain special characters.	No

8 Transaction Response (ACH)

The following table lists the variable names and requirements for each.

Transaction Response Variables for APPROVED Transactions (ACH)

Variable	Value	Description	Returned
\$cid		Returned of received data from merchant	Yes
\$transId		18 Character alphanumeric string as id for the transaction	Yes
\$appCode		6 Character alphanumeric string sent by card association as their transaction identifier.	Yes
\$actCode	000 – 999	3 Digit code to indicate the APPROVAL or DECLINE Code 000 = APPROVED 001 – 999 = DECLINE A full listing of decline codes in .xls format can be obtained by sending an email to assist@jetpay.com and request a listing of Decline Codes.	Yes
\$name \$fName \$lName		Name of customer returned in the format sent by the merchant	Yes

Variables Returned if sent by the Merchant (ACH)

Variable	Value
\$orderNum	As sent by the merchant
\$ud1	As sent by the merchant
\$ud2	As sent by the merchant
\$ud3	As sent by the merchant

9 Certification Testing

JetPay certification testing is intended to prepare a merchant for processing transactions through JetPay. This document contains test suites and test cases that enable a merchant to prepare and submit test JetPay transactions and verify compliance with JetPay submission rules.

9a Standard Transaction Processing Test Suite

The following test cases consist of common JetPay transactions that many merchants will need to be able to perform. To certify, the merchant needs to submit those test cases that apply to the merchant's anticipated requirements.

The following test cases will certify standard sales transactions. The credit card used for these test cases will follow the format, 4111 1111 1111 xxxx.

Test Case	Type	PAN	Exp Date	Amount	Expected Result
STND001	SALE	1111	12/13	\$10.00	Approved
STND002	SALE	1129	01/11	\$10.00	Expired Card
STND003	SALE	1128	12/13	\$10.00	Invalid Card Number
STND004 Through STND104	SALE Or AUTHONLY	1137	12/13	Amounts between \$1.00 and \$2.00 (\$1.01, \$1.02, etc)	Assorted responses both approved and declined. These test cases are always required.

9b CVV2 Test Cases

The credit card companies support multiple methods for validating credit cards and verifying a cardholder's identity. JetPay supports CVV2, CVC2, and CID card validation, available from Visa, MasterCard, and American Express, as well as all the other major credit card brands. JetPay supports address verification ("AVS") as implemented by Visa, MasterCard, and American Express (as well as Amex's name verification feature). With these two common features, a credit card's integrity can be validated and a cardholder's billing address can be verified. Even more importantly, these two features reduce transaction charges and help to protect a merchant against charge backs.

The following test cases will certify CVV2 transactions. The credit card used for these test cases use the following format: 4666 6666 6666 xxxx.

Test Case	Type	PAN	Exp Date	Amount	CVV2	Expected Result
CVV201	SALE Or AUTHONLY	6669	12/13	\$10.00	321	Approved, CVV Matches
CVV202	SALE Or AUTHONLY	6677	12/13	\$10.00	432	Approved, CVV Doesn't Match
CVV203	SALE Or	6685	12/13	\$10.00	543	Approved, CVV not processed

	AUTHONLY					
CVV204	AUTHONLY	6693	12/13	\$0.00	021	Approved, CVV Matches
CVV205	SALE or AUTHONLY	6610	12/13	\$10.05	123	Declined, CVV Not Processed
CVV206	SALE or AUTHONLY	6628	12/13	\$12.51	089	Declined, CVV Matches

9c Address Verification Test Cases

The following test cases will certify address verification transactions using AVS. The credit card used for these test cases is 4777 7777 7777 xxxx.

Test AVS if AVS is activated on the Merchant Account.

Test Case	Type	PAN	Exp Date	Amount	Address Postal Code	Expected Result
AVS01	SALE Or AUTHONLY	7711	01/11	\$11.00	1234 Fifth Street 77708	Approved, both address & zip match
AVS02	SALE Or AUTHONLY	7729	01/11	\$11.00	1234 Fifth Street 11100	Approved, address match, zip not match
AVS03	SALE Or AUTHONLY	7737	01/11	\$11.00	1234 Fifth Street 88809	Approved, zip match, address does not match
AVS04	SALE or AUTHONLY	7745	01/11	\$11.00	1234 Fifth Street 44403	Approved, address and zip do not match
AVS05	SALE or AUTHONLY	7778	01/11	\$11.00	1234 Fifth Street 33312	Approved, international non-AVS participant
AVS06	SALE or AUTHONLY	7810	01/11	\$12.51	1234 Fifth Street 00006	Declined, AVS Unavailable

9d Checking Account and Savings Account Transactions (ACH)

Merchants may submit bank drafts to JetPay. JetPay can enable properly configured merchants to send checks and savings account drafts directly into their bank account. By submitting the ABA routing code listed on every check along with the bank account number and the accountholder's name, JetPay will route these bank drafts directly into the merchant's bank account. Refunds of bank drafts are also available.

Bank accounts submitted through ACH require an ABA number, and account number, an accountholder's name, and an identifying check number. The following processes prescribe the test cases for ACH processing. For Approval Testing use routing code of 122000496, for Decline Testing use routing number of 654654654. For the cardholder name, use "John Q. Public" or the name of your favorite Star Wars character.

Test Case	Type	Account Number	Check Number	Amount	Expected Result
ACH01	CHECK	123411	234	\$10.00	Approved
ACH02	CHECK	552777	567	\$10.00	Approved
ACH03	CHECK	765432	890	\$10.00	Approved
ACH04	CHECK	3213211	12222	\$10.00	Decline
ACH05	CHECK	1598887	55622	\$10.00	Decline

APPENDIX

Visa AVS Result Codes (Credit Card)

Value	Definition
A	Street addresses match. The street addresses match but the postal/ZIP codes do not, or the request does not include the postal/ZIP code.
B	Street addresses match. Postal code not verified due to incompatible formats.(Acquirer sent both street address and postal code).
C	Street address and postal code not verified due to incompatible formats.(Acquirer sent both street address and postal code).
D	Street address and postal codes match.
F	Street address and postal ode match. Applies to U.K. Only.
G	Address information not verified for international transaction.
I	Address information not verified.
M	Street address and postal code match.
N	No match. Acquirer sent postal/ZIP code only, or street address only, or both postal code and street address.
P	Postal code match. Acquirer sent both postal code and street address, but street address not verified due to incompatible formats.
R	Retry: System unavailable or timed out. Issuer ordinarily performs their own AVS but was unavailable. Available for U.S. issuers only.
U	Address not verified for domestic transaction. Visa tried to perform check on issuer's behalf but no AVS information was available on record, issuer is not an AVS participant, or AVS data was present in the request but issuer did not return an AVS result.
Y	Street address and postal code match.
Z	Postal/ZIP matches; street does not match or street address not included in request.

Master Card AVS Result Codes (Credit Card)

Value	Definition
A	Address matches, postal code does not.
N	Neither address nor postal code matches.
R	Retry: System unavailable or timed out.
S	AVS currently not supported.
U	No data from issuer/Authorization System.
W	For U.S. addresses, nine-digit postal code matches, address does not, for address outside the U.S. postal code matches, address does not.
X	For U.S. addresses, nine-digit postal code and address matches, for address outside the U.S., postal code and address match.
Y	For U.S. addresses, five-digit postal code and address matches.
Z	For U.S. addresses, five-digit postal code matches, address does not.

URL Generator Credit Card (sample in PHP)

```
<?php
//File Name url_generator.php

//Used to start PHP
session_start();

//STRIP UNWANTED CHARICTORS FROM INPUT
function StripSpecChar($val) {
    return (preg_replace('/[^a-zA-Z0-9" "-.@\:\/_]/',' ', $val));
}

// USE EACH POSTED VALUE AND CHANGE TO SESSION DATA FOR XML
if (isset($_POST['transType'])) {
    foreach ($_POST as $key => $val) {
        $_SESSION[$key] = StripSpecChar($val);
    }
} else {
    foreach ($_REQUEST as $key => $val) {
        $_SESSION[$key] = StripSpecChar($val);
    }
}

// REQUIRED FOR XSS SCRIPTING WORK AROUND DO NOT EDIT
header('P3P: CP="CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR"');

//Merchant Varriables for URL*****
$tid = "TESTTERMINAL";

// JETPAY IFRAME URL*****
$jpUrl = " "; //TESTING URL
//$ jpUrl = "https"; //LIVE URL

//MERCHANTS CAN NOT SEND MORE THAN TRANSACTION TYPE GROUP OR THE TRANSACTION WILL BE DECLINED!

//SALE ONLY MERCHANT WITH TOKENIZATION
//$transType = "&transType=" . rawurlencode("SALE");
//$verSub = "&verSub=" . rawurlencode("JL4.10SATO");
//$tokenTrans = "&tokenTrans=" . rawurlencode("true");

//AVS*****
$getAvs = "true";
```

```

//CSS URL*****
$css = https://link_to_jetlink_css_sheet_here.com";

//APPROVED RETURN URL*****
$retUrl = "https://your_approved_return_url_here.com";

//DECLINE RETURN URL*****
$decUrl = "https://your_decline_url_here.com";

//CANCEL URL*****
$cancelUrl = "https://your_cancel_url_here.com";

GET USERS EMAIL ADDRES*****
$getEmail = "true";

//Set Session ID for CID if not provided.*****
if ($cid == "") {
    $callId = session_id();
    $cid = "cid=" . rawurlencode($callId);
}

//BUILD URL TO CONNECT TO JETPAY JETLINK*****
//CORE MINIMUM FOR TRANSACTION*****
$strUrl = $jpUrl;
$strUrl .= "?" . $cid;
$strUrl .= "&tid=" . rawurlencode($tid);
$strUrl .= $transType;
$strUrl .= $verSub;
if ($tokenTrans != "") {
    $strUrl .= $tokenTrans;
}

if ($transType != "TOKENIZE") {
    $strUrl .= "&amount=" . rawurlencode($_SESSION['amount']);
}

$strUrl .= "&retUrl=" . rawurlencode($retUrl);
$strUrl .= "&decUrl=" . rawurlencode($decUrl);
$strUrl .= "&cancelUrl=" . rawurlencode($cancelUrl);

if ($_SESSION['name'] != "") {
    $strUrl .= "&name=" . rawurlencode($_SESSION['name']);
}

if ($_SESSION['fName'] && $_SESSION['lName'] != "") {
    $strUrl .= "&fName=" . rawurlencode($_SESSION['fName']);
    $strUrl .= "&lName=" . rawurlencode($_SESSION['lName']);
}

```

```

}

//OPTIONAL CSS URL VARIABLE
if ($css != "") {
    $strUrl .= "&css=" . rawurlencode($css);
}

//EMAIL
//OPTIONAL EMAIL VARIABLE
if ($getEmail == "true" && $_SESSION['email'] == "") {
    $strUrl .= "&getEmail=" . rawurlencode($getEmail);
} else {
    if ($_SESSION['email'] != "") {
        $strUrl .= "&email=" . rawurlencode($_SESSION['email']);
    }
}

//OPTIONAL AVS INFORMATION
if ($getAvs == "false") {
    $strUrl .= "&address1=" . rawurlencode($_SESSION['address1']);
    if ($_SESSION['address2'] != "") {
        $strUrl .= "&address2=" . rawurlencode($_SESSION['address2']);
    }
}

$strUrl .= "&city=" . rawurlencode($_SESSION['city']);
$strUrl .= "&state=" . rawurlencode($_SESSION['state']);
$strUrl .= "&zip=" . rawurlencode($_SESSION['zip']);

if ($_SESSION['country'] != "") {
    $strUrl .= "&country=" . rawurlencode($_SESSION['country']);
} else {
    if ($getAvs == "true" || $getAvs == "none") {
        $strUrl .= "&getAvs=" . rawurlencode($getAvs);
    }
}

//OPTIONAL VARIABLE FIELDS
if ($_SESSION['orderNum'] != "") {
    $strUrl .= "&orderNum=" . rawurlencode($_SESSION['orderNum']);
}

if ($_SESSION['ud1'] != "") {
    $strUrl .= "&ud1=" . rawurlencode($_SESSION['ud1']);
}

if ($_SESSION['ud2'] != "") {
    $strUrl .= "&ud2=" . rawurlencode($_SESSION['ud2']);
}

```

```
}  
if ($_SESSION['ud3'] != "") {  
    $strUrl .= "&ud3=" . rawurlencode($_SESSION['ud3']);  
}  
//CLOSE URL  
$strUrl .= "&urlEnd=" . rawurlencode("END");  
$jpurl = $strUrl;  
?>
```



```

    }

//CSS URL*****
$css = https://link_to_jetlink_css_sheet_here.com";

//APPROVED RETURN URL*****
$retUrl = "https://your_approved_return_url_here.com";

//DECLINE RETURN URL*****
$decUrl = "https://your_decline_url_here.com";

//CANCEL URL*****
$cancelUrl = "https://your_cancel_url_here.com";
GET USERS EMAIL ADDRES*****
$getEmail = "true";

//Set Session ID for CID if not provided.*****
if ($cid == "") {
    $callId = session_id();
    $cid = "cid=" . rawurlencode($callId);
}

//BUILD URL TO CONNECT TO JETPAY JETLINK*****
//CORE MINIMUM FOR TRANSACTION*****
$strUrl = $jpUrl;
$strUrl .= "?" . $cid;
$strUrl .= "&tid=" . rawurlencode($tid);
$strUrl .= "&transType=" . rawurlencode($transType);
$strUrl .= "&sec=" . rawurlencode($sec);
$strUrl .= "&verSub=" . rawurlencode($verSub);
$strUrl .= "&amount=" . rawurlencode($_SESSION['amount']);
$strUrl .= "&retUrl=" . rawurlencode($retUrl);
$strUrl .= "&decUrl=" . rawurlencode($decUrl);
if ($cancelUrl != "") {
    $strUrl .= "&cancelUrl=" . rawurlencode($cancelUrl);
}

//OPTIONAL EMAIL DATA*****
if ($getEmail == "true" && $_SESSION['email'] == "") {
    $strUrl .= "&getEmail=" . rawurlencode($getEmail);
} else {
    if ($_SESSION['email'] != "") {
        $strUrl .= "&email=" . rawurlencode($_SESSION['email']);
    }
}

//OPTIONAL DATA*****
if ($_SESSION['name'] != "") {

```

```
        $strUrl .= "&name=" . rawurlencode($_SESSION['name']);
    }
    if ($_SESSION['orderNum'] != "") {
        $strUrl .= "&orderNum=" . rawurlencode($_SESSION['orderNum']);
    }
    if ($_SESSION['ud1'] != "") {
        $strUrl .= "&ud1=" . rawurlencode($_SESSION['ud1']);
    }
    if ($_SESSION['ud2'] != "") {
        $strUrl .= "&ud2=" . rawurlencode($_SESSION['ud2']);
    }
    if ($_SESSION['ud3'] != "") {
        $strUrl .= "&ud3=" . rawurlencode($_SESSION['ud3']);
    }

    $strUrl .= "&urlEnd=" . rawurlencode("END");

    $jpurl = $strUrl;

?>
```

Sample HTML for setting <iframe> tag

```
<?php
    session_start();
    include_once('../includes/url_generator.php');
    header('P3P: CP="CURa ADMa DEVa PSAo PSDo OUR BUS UNI PUR INT DEM STA PRE COM NAV OTC NOI DSP COR"');
?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Merchant iFrame Page</title>
</head>

<body>
<div align="center">
<iframe src="<?PHP print $jpurl; ?>" width="550px" height="800px" frameborder="0"></iframe>
</div>

</body>
</html>
```

Country List

Sample three(3) character to full name country list in and associative array.

```
$ISO3166ThreeToName = array(  
    'AFG' => 'Afghanistan',  
    'ALB' => 'Albania',  
    'DZA' => 'Algeria',  
    'ASM' => 'American Samoa',  
    'AND' => 'Andorra',  
    'AGO' => 'Angola ',  
    'AIA' => 'Anguilla',  
    'ATA' => 'Antarctica',  
    'ATG' => 'Antigua and Barbuda',  
    'ARG' => 'Argentina Argentine ',  
    'ARM' => 'Armenia',  
    'ABW' => 'Aruba',  
    'AUS' => 'Australia',  
    'AUT' => 'Austria',  
    'AZE' => 'Azerbaijan',  
    'BHS' => 'Bahamas',  
    'BHR' => 'Bahrain',  
    'BGD' => 'Bangladesh',  
    'BRB' => 'Barbados',  
    'BLR' => 'Belarus',  
    'BEL' => 'Belgium',  
    'BLZ' => 'Belize',  
    'BEN' => 'Benin',  
    'BMU' => 'Bermuda',  
    'BTN' => 'Bhutan',  
    'BOL' => 'Bolivia',  
    'BIH' => 'Bosnia/Herzegovina',  
    'BWA' => 'Botswana ',  
    'BVT' => 'Bouvet Island',  
    'BRA' => 'Brazil',  
    'IOT' => 'British Indian Ocean Territory',  
    'VGB' => 'British Virgin Islands',  
    'BRN' => 'Brunei Darussalam',  
    'BGR' => 'Bulgaria',  
    'BFA' => 'Burkina Faso',  
    'BDI' => 'Burundi ',  
    'KHM' => 'Cambodia',  
    'CMR' => 'Cameroon ',  
    'CAN' => 'Canada',  
    'CPV' => 'Cape Verde',  
    'CYM' => 'Cayman Islands',  
    'CAF' => 'Central African ',  
    'TCD' => 'Chad',  
    'CHL' => 'Chile',  
    'CHN' => 'China',  
    'CXR' => 'Christmas Island',  
    'CCK' => 'Cocos (Keeling) Islands',  
    'COL' => 'Colombia',  
    'COM' => 'Comoros',  
    'COD' => 'Congo Democratic',  
    'COG' => 'Congo',  
    'COK' => 'Cook Islands',  
    'CRI' => 'Costa Rica',  
    'CIV' => 'Ivory Coast',  
    'CUB' => 'Cuba',  
    'CYP' => 'Cyprus',  
    'CZE' => 'Czech ',  
    'DNK' => 'Denmark',  
    'DJI' => 'Djibouti',  
    'DMA' => 'Dominica',  
    'DOM' => 'Dominican ',  
    'TLS' => 'Timor-Leste',  
    'ECU' => 'Ecuador',  
    'EGY' => 'Egypt Arab',
```

'SLV' => 'El Salvador',
'GNQ' => 'Equatorial Guinea',
'ERI' => 'Eritrea',
'EST' => 'Estonia',
'ETH' => 'Ethiopia',
'FRO' => 'Faeroe Islands',
'FLK' => 'Falkland Islands',
'FJI' => 'Fiji',
'FIN' => 'Finland',
'FRA' => 'France',
'GUF' => 'French Guiana',
'PYF' => 'French Polynesia',
'ATF' => 'French Southern Territories',
'GAB' => 'Gabon Gabonese ',
'GMB' => 'Gambia the',
'GEO' => 'Georgia',
'DEU' => 'Germany',
'GHA' => 'Ghana',
'GIB' => 'Gibraltar',
'GRC' => 'Greece Hellenic ',
'GRL' => 'Greenland',
'GRD' => 'Grenada',
'GLP' => 'Guadeloupe Islands',
'GUM' => 'Guam',
'GTM' => 'Guatemala',
'GIN' => 'Guinea',
'GNB' => 'Guinea-Bissau',
'GUY' => 'Guyana',
'HTI' => 'Haiti',
'HMD' => 'Heard & McDonald Islands',
'VAT' => 'Holy See (Vatican City)',
'HND' => 'Honduras',
'HKG' => 'Hong Kong',
'HRV' => 'Croatia',
'HUN' => 'Hungary',
'ISL' => 'Iceland',
'IND' => 'India',
'IDN' => 'Indonesia',
'IRN' => 'Iran Islamic',
'IRQ' => 'Iraq',
'IRL' => 'Ireland',
'ISR' => 'Israel',
'ITA' => 'Italy',
'JAM' => 'Jamaica',
'JPN' => 'Japan',
'JOR' => 'Jordan',
'KAZ' => 'Kazakhstan',
'KEN' => 'Kenya',
'KIR' => 'Kiribati',
'PRK' => 'Korea',
'KOR' => 'Korea',
'KWT' => 'Kuwait',
'KGZ' => 'Kyrgyz ',
'LAO' => 'Lao',
'LVA' => 'Latvia',
'LBN' => 'Lebanon',
'LSO' => 'Lesotho',
'LBR' => 'Liberia',
'LBY' => 'Libya',
'LIE' => 'Liechtenstein',
'LTU' => 'Lithuania',
'LUX' => 'Luxembourg',
'MAC' => 'Macao',
'MKD' => 'Macedonia',
'MDG' => 'Madagascar',
'MWI' => 'Malawi',
'MYS' => 'Malaysia',
'MDV' => 'Maldives',
'MLI' => 'Mali',
'MLT' => 'Malta',
'MHL' => 'Marshall Islands',
'MTQ' => 'Martinique',

'MRT' => 'Mauritania',
 'MUS' => 'Mauritius',
 'MYT' => 'Mayotte',
 'MEX' => 'Mexico',
 'FSM' => 'Micronesia',
 'MDA' => 'Moldova',
 'MCO' => 'Monaco',
 'MNG' => 'Mongolia',
 'MSR' => 'Montserrat',
 'MAR' => 'Morocco',
 'MOZ' => 'Mozambique',
 'MMR' => 'Myanmar',
 'NAM' => 'Namibia',
 'NRU' => 'Nauru',
 'NPL' => 'Nepal',
 'ANT' => 'Netherlands Antilles',
 'NLD' => 'Netherlands',
 'NCL' => 'New Caledonia',
 'NZL' => 'New Zealand',
 'NIC' => 'Nicaragua',
 'NER' => 'Niger the',
 'NGA' => 'Nigeria',
 'NIU' => 'Niue',
 'NFK' => 'Norfolk Island',
 'MNP' => 'Northern Mariana Islands',
 'NOR' => 'Norway',
 'OMN' => 'Oman',
 'PAK' => 'Pakistan',
 'PLW' => 'Palau',
 'PSE' => 'Palestinian Territory',
 'PAN' => 'Panama',
 'PNG' => 'Papua New Guinea',
 'PRY' => 'Paraguay',
 'PER' => 'Peru',
 'PHL' => 'Philippines',
 'PCN' => 'Pitcairn Island',
 'POL' => 'Poland Polish ',
 'PRT' => 'Portugal',
 'PRI' => 'Puerto Rico',
 'QAT' => 'Qatar State',
 'REU' => 'Reunion',
 'ROU' => 'Romania',
 'RUS' => 'Russia',
 'RWA' => 'Rwanda',
 'SHN' => 'St. Helena',
 'KNA' => 'St. Kitts & Nevis',
 'LCA' => 'St. Lucia',
 'SPM' => 'St. Pierre & Miquelon',
 'VCT' => 'St. Vincent/Grenadines',
 'WSM' => 'Samoa',
 'SMR' => 'San Marino',
 'STP' => 'Sao Tome',
 'SAU' => 'Saudi Arabia',
 'SEN' => 'Senegal',
 'SYC' => 'Seychelles',
 'SLE' => 'Sierra Leone',
 'SGP' => 'Singapore',
 'SVK' => 'Slovakia',
 'SVN' => 'Slovenia',
 'SLB' => 'Solomon Islands',
 'SOM' => 'Somalia Somali ',
 'ZAF' => 'South Africa',
 'SGS' => 'South Georgia',
 'SGS' => 'South Sandwich Islands',
 'ESP' => 'Spain',
 'LKA' => 'Sri Lanka',
 'SDN' => 'Sudan',
 'SUR' => 'Suriname',
 'SJM' => 'Svalbard & Jan Mayen Islands',
 'SWZ' => 'Swaziland',
 'SWE' => 'Sweden',
 'CHE' => 'Switzerland',

```
'SYR' => 'Syrian',
'TWN' => 'Taiwan',
'TJK' => 'Tajikistan',
'TZA' => 'Tanzania',
'THA' => 'Thailand',
'TGO' => 'Togo Togolese ',
'TKL' => 'Tokelau',
'TON' => 'Tonga',
'TTO' => 'Trinidad & Tobago',
'TUN' => 'Tunisia',
'TUR' => 'Turkey',
'TKM' => 'Turkmenistan',
'TCA' => 'Turks & Caicos Islands',
'TUV' => 'Tuvalu',
'VIR' => 'Virgin Islands',
'UGA' => 'Uganda',
'UKR' => 'Ukraine',
'ARE' => 'United Arab Emirates',
'GBR' => 'United Kingdom',
'UMI' => 'US Minor Outlying Islands',
'USA' => 'United States of America',
'URY' => 'Uruguay',
'UZB' => 'Uzbekistan',
'VUT' => 'Vanuatu',
'VEN' => 'Venezuela',
'VNM' => 'Viet Nam',
'WLF' => 'Wallis & Futuna Islands',
'ESH' => 'Western Sahara',
'YEM' => 'Yemen',
'YUG' => 'Yugoslavia',
'ZMB' => 'Zambia',
'ZWE' => 'Zimbabwe',
);
```