Website Payments Plus Integration Guide

Japan

Last updated: May 2014
## Contents

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preface</td>
<td>.</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>About This Guide</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Intended Audience</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Revision History</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Documentation Feedback</td>
<td>9</td>
</tr>
<tr>
<td>Chapter 1</td>
<td>Getting Started with Website Payments Plus</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Overview</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Features and Benefits</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>How Website Payments Plus Works</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>PayPal Express Checkout Transaction Processing</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td>Introduction to Integrating with Website Payments Plus</td>
<td>13</td>
</tr>
<tr>
<td>Chapter 2</td>
<td>Integrating Your Website Using HTML</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Simple Website Payments Plus Integration</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>Sample Integration</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>HTML Variables for Payment Page Settings</td>
<td>17</td>
</tr>
<tr>
<td>Chapter 3</td>
<td>Customising Your PayPal Payment Page</td>
<td>21</td>
</tr>
<tr>
<td></td>
<td>Modifying Your PayPal Account Settings</td>
<td>22</td>
</tr>
<tr>
<td></td>
<td>Settings</td>
<td>23</td>
</tr>
<tr>
<td></td>
<td>Customisation</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>The Mobile Optimized Payment Flow</td>
<td>36</td>
</tr>
<tr>
<td></td>
<td>HTML Variables for Payment Page Look and Feel</td>
<td>38</td>
</tr>
<tr>
<td>Chapter 4</td>
<td>Integrating iFrame in Your Website</td>
<td>41</td>
</tr>
<tr>
<td></td>
<td>Integrating iFrame</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>Manual Integration</td>
<td>42</td>
</tr>
<tr>
<td></td>
<td>API Integration</td>
<td>44</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>Integrating Your Website Using API</td>
<td>47</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------</td>
<td></td>
</tr>
<tr>
<td>GetTransactionDetails Response</td>
<td>72</td>
<td></td>
</tr>
<tr>
<td>RefundTransaction API</td>
<td>81</td>
<td></td>
</tr>
<tr>
<td>RefundTransaction Request</td>
<td>81</td>
<td></td>
</tr>
<tr>
<td>RefundTransaction Response</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>DoCapture API</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>DoCapture Request</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>DoCapture Response</td>
<td>85</td>
<td></td>
</tr>
</tbody>
</table>

**Appendix B  Moving from Website Payments Standard to Website Payments Plus** | 89

Sample Code Comparison. | 90

**Appendix C  Error Messages** | 93

**Appendix D  Currency Codes** | 95
About This Guide

The Website Payments Plus Integration Guide describes how to integrate with Website Payments Plus. It includes information about:

- Features and benefits of Website Payments Plus.
- Moving from Website Payments Standard to Website Payments Plus.
- Integrating your website with Website Payments Plus.
- Customising your hosted payment page.
- Testing your integration in the Sandbox environment.
- Verifying the status and authenticity of the transactions before fulfilling the orders.

Intended Audience

This guide is for merchants and developers that want to integrate with Website Payments Plus to add transaction processing to their website.

Website Payments Plus is not available in the United States. It is only available in the countries in the table below. The product name may vary from country to country.

<table>
<thead>
<tr>
<th>Country</th>
<th>Product Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Australia</td>
<td>Website Payments Pro Hosted Solution</td>
</tr>
<tr>
<td>France</td>
<td>PayPal Intégral Evolution</td>
</tr>
<tr>
<td>Hong Kong</td>
<td>Website Payments Pro Hosted Solution</td>
</tr>
<tr>
<td>Singapore</td>
<td>Website Payments Pro Hosted Solution</td>
</tr>
<tr>
<td>Italy</td>
<td>PayPal Pro</td>
</tr>
<tr>
<td>Japan</td>
<td>Website Payments Plus</td>
</tr>
<tr>
<td>Spain</td>
<td>Pasarela Integral</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>Website Payments Pro Hosted Solution</td>
</tr>
</tbody>
</table>

For more information go to
## Revision History

Revision history for *Website Payments Plus Integration Guide*.

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>May 2014</td>
<td>Updated the secure payments URL shown in the examples throughout the guide. Removed the s-xclick command from all examples and also updated the example hosted button IDs to use HSSS instead of HSS.</td>
</tr>
<tr>
<td>October 2013</td>
<td>Updated the list of supported browsers in <em>Integrating iFrame in Your Website</em>.</td>
</tr>
<tr>
<td>August 2013</td>
<td>Updated the <em>Testing Your Integration in Sandbox</em> and <em>Error Messages</em> chapters. Updated the possible values of the language code lc request field in the <em>HTML Variables for Payment Page Settings</em> section. Added a note in the <em>Verifying Transaction Status and Authenticity</em> section on the PayPal account settings which result in appending each Transaction ID to your return URL. Updated content and URLs.</td>
</tr>
<tr>
<td>February 2013</td>
<td>Added a section on <em>The Mobile Optimized Payment Flow</em>. Updated the values of the template HTML variable in the <em>HTML Variables for Payment Page Look and Feel</em> table. Added information on mobile optimized payment pages in the <em>Integrating iFrame in Your Website</em> chapter.</td>
</tr>
<tr>
<td>October 2012</td>
<td>Added the new payment page customisation features to the <em>Customising Your PayPal Payment Page</em> chapter.</td>
</tr>
<tr>
<td>August 2012</td>
<td>Updated the <em>Intended Audience</em> section. Updated the endpoint in the examples.</td>
</tr>
<tr>
<td>July 2012</td>
<td>Removed references to the deprecated HTML variable: shopping_url</td>
</tr>
<tr>
<td>June 2012</td>
<td>• Updated the requirements for the billing address fields in the <em>HTML Variables for Payment Page Settings</em> section.</td>
</tr>
<tr>
<td></td>
<td>• Added a character limitations note to the following sections: <em>HTML Variables for Payment Page Settings</em>, <em>HTML Variables for Payment Page Look and Feel</em> and <em>BMCreateButton API Operation</em>.</td>
</tr>
<tr>
<td></td>
<td>• Corrected the HTML samples in <em>Chapter 2</em> and <em>Chapter 5</em>.</td>
</tr>
<tr>
<td>May 2012</td>
<td>Added IE9 to list of browsers that support iFrame.</td>
</tr>
</tbody>
</table>
Documentation Feedback

Help us improve this guide by sending feedback to:

documentationfeedback@paypal.com
Overview

Website Payments Plus is a payment platform allowing merchants to receive payments funded by card or PayPal account. This solution is hosted by PayPal. You do not have to capture or store credit card information on your website, thereby helping towards achieving PCI compliance.

Website Payments Plus is the choice for merchants who prefer a solution where all financial details are handled by PayPal.

In addition to Website Payments Plus, PayPal recommends that you implement the PayPal Express Checkout button on your website. The button appears much earlier in the payment flow and gives existing PayPal account holders the opportunity to use PayPal, thereby increasing the transaction completion rate.

Features and Benefits

Here are the features and benefits of Website Payments Plus:

- **PCI compliance** - Payment Card Industry (PCI) Data Security Standards (DSS) is a global security standard which applies to all businesses that collect, store, process, or transmit card holder information. You can use PayPal’s hosted payment page as part of the PCI compliance solution for your business.

  **Note:** Website Payments Plus implementation helps achieving PCI compliance, and does not necessarily guarantee it.

- **Supports iFrame** - PayPal offers a compact payment form that can be integrated in an iFrame on your website. The buyer completes the payment on your website and you can maintain the checkout look and feel in the master frame that surrounds the compact form. The credit card fields are part of the compact form so you do not have to collect this information separately. For more information, refer to “Integrating iFrame in Your Website” on page 41.

- **Virtual Terminal** - PayPal-hosted online payment form, which enables you to accept phone fax, mail orders for all major credit cards.

- **PayPal Express Checkout** - Two-click payment option for PayPal account holders. For more information, refer to “PayPal Express Checkout Transaction Processing” on page 13.

- **Supports Major Credit and Debit Cards** - Supports Visa, MasterCard, JCB.
Getting Started with Website Payments Plus

How Website Payments Plus Works

- **Mobile Support** - Website Payments Plus payment pages are optimized for both desktop and mobile browsers. For more information, refer to “The Mobile Optimized Payment Flow” on page 36.

---

How Website Payments Plus Works

In the figure above, the top flow is for paying using your PayPal account and the bottom flow is for paying with a card.

To integrate your website with Website Payments Plus:

1. Generate a button in your website checkout flow using HTML or API solution. The button could be labeled **Pay** or **Buy** or similar. When the buyer clicks this button, they are redirected to the payment page hosted by PayPal.

2. On the payment page, the buyer enters their debit or credit card information and clicks the **Pay Now** button. They do not have to have a PayPal account, although they have the option to use it if they have one.

3. If the transaction is successful, the buyer either sees PayPal’s confirmation page or is redirected to a URL you specify. If the transaction is unsuccessful, an error message is displayed, and the buyer can rectify the error and retry the transaction.

You can specify the content of the payment page and configure its appearance to reflect the look and feel of your website (including your logo).
PayPal Express Checkout Transaction Processing

PayPal Express Checkout is PayPal’s premier checkout solution. It makes it easier for your buyers to pay and allows you to accept PayPal while retaining control of the buyer and overall checkout flow. It enables the buyers to use shipping information stored securely at PayPal to check out, so they do not have to re-enter it on your site. The **Pay with PayPal** button can be placed on both the product page as well as the shopping cart page and is a great tool to increase checkout conversion.

The following steps describe how PayPal Express Checkout works

1. After selecting products to purchase on your website, buyers click **Pay with PayPal** button.
2. They are redirected to the PayPal site where they log in using their PayPal login and password.
3. After logging in, they review the order summary, and click **Continue Checkout** to authorise the transaction.
4. The buyers are then returned to the confirmation page on your website where they can verify the order details and submit the transaction.

For complete details on PayPal Express Checkout, refer to the [Express Checkout Integration Guide](#).

Introduction to Integrating with Website Payments Plus

After signing up for Website Payments Plus, follow these steps to integrate your website with:

1. Connect your website to Website Payments Plus: Identify a point in your website checkout flow where you want to place a **Pay** or similar button that the buyer clicks on to initiate the payment. You can do this using HTML or API. Clicking on this button redirects the buyer’s browser to PayPal’s payment page for transaction processing. Depending on how you want
to integrate, follow the steps described in Chapter 2, “Integrating Your Website Using HTML” on page 15 or Chapter 5, “Integrating Your Website Using API” on page 47.

2. Optionally, customise the appearance and content of your PayPal-hosted payment page by either sending the appropriate HTML variables or by editing your PayPal account settings. Refer to Chapter 3, “Customising Your PayPal Payment Page” on page 21.

3. Optionally, test your integration in the PayPal Sandbox environment. This step is described in Chapter 6, “Testing Your Integration in Sandbox” on page 55.
This chapter provides instructions for a simple integration that enables you to begin processing transactions using Website Payments Plus.

**Note:** PayPal recommends that you implement the simple integration to familiarise yourself with Website Payments Plus before implementing a more customised integration.

As part of a simple integration, you get the default settings on your payment page. To customise the look and feel of the page so it matches your website, you can do one of the following:

- Change your settings in your Profile section on PayPal.com, as described in “Modifying Your PayPal Account Settings” on page 22.
- Add the appropriate HTML variables to the payment page, as described in Table 2.1, “HTML Variables for Settings of Payment Page” on page 17 and Table 3.2, “HTML Variables for Look and Feel of Payment Page” on page 38.

**Important:** HTML variables will override the settings you save on your profile page.

---

### Simple Website Payments Plus Integration

To integrate your website with Website Payments Plus, identify a point in your website checkout flow where you want to place a button that the buyer clicks to initiate the payment. The button should be labeled **Continue to Payment**, **Pay** or similar, and when clicked, should execute a Form POST to PayPal. Clicking on this button redirects the buyer’s browser to the PayPal payment page where they can pay with credit card, or their PayPal account.

The Form POST contains a set of HTML variables that describe the transaction. In the Form POST, you must specify the following:

- **subtotal** - amount of the transaction
- **business** - Secure Merchant ID (found on the profile page) or the email address associated with your PayPal account.
- **paymentaction** - Indicates whether the transaction is for payment on a final sale or an authorisation for a final sale (to be captured later).

The default currency is USD. Additionally, you can specify the appropriate HTML variables listed in Table 2.1, “HTML Variables for Settings of Payment Page” on page 17 to customise the information collected on the payment page or Table 3.2, “HTML Variables for Look and Feel of Payment Page” on page 38 to customise the look and feel of the page. If the payment is successful, then the buyer either sees the PayPal confirmation page or is redirected to the URL you specify in your configuration.
The return URL is appended with a Transaction ID on the query string during the redirect back from the payment page to your website. This Transaction ID can be used to retrieve the status and verify the authenticity of the transaction. For detailed information on verifying the authenticity of the transaction before fulfilling the order, refer to Chapter 7, “Order Processing” on page 61.

Sample Integration

Below is an example of a simple Website Payments Plus integration:

1. Sample Website Payments Plus Form POST:

   ```html
   <input type="hidden" name="cmd" value="_hosted-payment">
   <input type="hidden" name="subtotal" value="50">
   <input type="hidden" name="business" value="HNZ3QZMCPBAAA">
   <input type="hidden" name="paymentaction" value="sale">
   <input type="hidden" name="return" value="https://yourwebsite.com/receipt_page.html">
   <input type="submit" name="METHOD" value="Pay">
   </form>
   ``

   The bold text is the value for the corresponding variable. It is recommended that you enclose the values in quotes. For detailed information on these values, refer to Table 2.1, “HTML Variables for Settings of Payment Page” on page 17.

2. Output the HTML text into your website at the point where buyers will proceed with their checkout.

3. Open your checkout page and test the button to ensure that it opens the PayPal payment page.

   You can also use the PayPal sandbox environment to test your integration. For complete information on testing your integration in the PayPal Sandbox environment, refer to Chapter 6, “Testing Your Integration in Sandbox” on page 55.
## HTML Variables for Payment Page Settings

The table below lists the Website Payments Plus HTML variables you can use to send in additional transaction information along with your web request. For a list of HTML variables that you can use to customise the look and feel of your payment page, refer to Table 3.2, “HTML Variables for Look and Feel of Payment Page” on page 38.

**Note:** The values you pass must not contain any of these special characters (}{<>\";

**Note:** Some merchants are required to pass billing information with every transaction. It is recommended that you test your integration first, particularly if you plan to use iFrame, to determine if the billing information fields are required.

### Table 2.1 HTML Variables for Settings of Payment Page

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>address1</td>
<td>Street name of shipping address. (1 of 2 fields).</td>
<td>No</td>
</tr>
<tr>
<td>address2</td>
<td>Street name of shipping address. (2 of 2 fields).</td>
<td>No</td>
</tr>
<tr>
<td>address_override</td>
<td>The payer is shown the passed-in address but cannot edit it. This variable is overridden if there are errors in the address. The allowable values are true/false. Default is false.</td>
<td>No</td>
</tr>
<tr>
<td>billing_address1</td>
<td>Street name of the billing address. (1 of 2 fields).</td>
<td>Conditional</td>
</tr>
<tr>
<td>billing_address2</td>
<td>Street name of the billing address. (2 of 2 fields).</td>
<td>No</td>
</tr>
<tr>
<td>billing_city</td>
<td>City name of the billing address.</td>
<td>Conditional</td>
</tr>
<tr>
<td>billing_country</td>
<td>Country code of the billing address.</td>
<td>Conditional</td>
</tr>
<tr>
<td>billing_first_name</td>
<td>First name of person the item is being billed to.</td>
<td>Conditional</td>
</tr>
<tr>
<td>billing_last_name</td>
<td>Last name of person the item is being billed to.</td>
<td>Conditional</td>
</tr>
<tr>
<td>billing_state</td>
<td>State name of the billing address.</td>
<td>Conditional</td>
</tr>
<tr>
<td>billing_zip</td>
<td>Zip code of the billing address.</td>
<td>Conditional</td>
</tr>
<tr>
<td>bn</td>
<td>Identifies the source that built the code for the button.</td>
<td>No</td>
</tr>
<tr>
<td><strong>Format</strong></td>
<td>- &lt;Company&gt;<em>&lt;Service&gt;</em>&lt;Product&gt;_&lt;Country&gt;</td>
<td></td>
</tr>
<tr>
<td>business</td>
<td>Your PayPal account email address or your PayPal ID (Secure Merchant ID) associated with your PayPal account. It is recommended that you use your PayPal ID, which can be found on the top section of the Profile page on PayPal.com</td>
<td>Yes</td>
</tr>
<tr>
<td>buyer_email</td>
<td>Email address of the buyer.</td>
<td>No</td>
</tr>
</tbody>
</table>
### Table 2.1 HTML Variables for Settings of Payment Page (Continued)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
<th>Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>cancel_return</td>
<td>The browser will be redirected to this URL if the buyer clicks “Return to Merchant” link. Be sure to enter the complete URL, including http:// or https://.</td>
<td>No</td>
</tr>
<tr>
<td>cbt</td>
<td>Sets the text for the “Return to Merchant” link on the PayPal confirmation page. For business accounts, the return button displays your business name in the place of the word “Merchant” by default.</td>
<td>No</td>
</tr>
<tr>
<td>city</td>
<td>City name of shipping address.</td>
<td>No</td>
</tr>
<tr>
<td>country</td>
<td>Country name of shipping address.</td>
<td>No</td>
</tr>
<tr>
<td>currency_code</td>
<td>The currency of the payment. The default is USD.</td>
<td>No</td>
</tr>
<tr>
<td>custom</td>
<td>Pass through variable never presented to the payer.</td>
<td>No</td>
</tr>
<tr>
<td>first_name</td>
<td>First name of person the item is being shipped to.</td>
<td>No</td>
</tr>
<tr>
<td>handling</td>
<td>Handling charged. This amount is added to subtotal for the total amount.</td>
<td>No</td>
</tr>
<tr>
<td>invoice</td>
<td>Order number in the merchant’s ordering/invoice system.</td>
<td>No</td>
</tr>
<tr>
<td>last_name</td>
<td>Last name of person the item is being shipped to.</td>
<td>No</td>
</tr>
<tr>
<td>lc</td>
<td>The display language of the login or sign-up page.</td>
<td>No</td>
</tr>
<tr>
<td>night_phone_a</td>
<td>The area code of the U.S. phone number, or the country code of the phone number outside the U.S. This pre-populates the buyer’s home phone number.</td>
<td>No</td>
</tr>
<tr>
<td>night_phone_b</td>
<td>The three-digit prefix for U.S. phone numbers, or the entire non-U.S. phone number for numbers outside the U.S., excluding the country code. This pre-populates the buyer’s home phone number. <strong>NOTE:</strong> Use this variable for non-US numbers.</td>
<td>No</td>
</tr>
<tr>
<td>night_phone_c</td>
<td>The four-digit phone number for U.S. phone numbers. This pre-populates the buyer’s home phone number.</td>
<td>No</td>
</tr>
<tr>
<td>notify_url</td>
<td>The URL to which PayPal posts information about the transaction in the form of Instant Payment Notification. Be sure to enter the complete URL, including http:// or https://.</td>
<td>No</td>
</tr>
<tr>
<td>paymentaction</td>
<td>Indicates whether the transaction is for payment on a final sale or an authorisation for a final sale (to be captured later). <strong>Allowable Values:</strong> - authorization or sale <strong>Default Value</strong> - sale</td>
<td>Yes</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
<td>Required</td>
</tr>
<tr>
<td>----------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>return</td>
<td>The URL to which the buyer’s browser is redirected to after completing the payment. Be sure to enter the complete URL, including http:// or https://.</td>
<td>No</td>
</tr>
<tr>
<td>shipping</td>
<td>Shipping charged. This amount is added to subtotal for the total amount.</td>
<td>No</td>
</tr>
<tr>
<td>state</td>
<td>State of the shipping address.</td>
<td>No</td>
</tr>
<tr>
<td>subtotal</td>
<td>Amount charged for the transaction. If shipping, handling, and taxes are not specified, this is the total amount charged.</td>
<td>Yes</td>
</tr>
<tr>
<td>tax</td>
<td>Taxes charged. This amount is added to subtotal for the total amount.</td>
<td>No</td>
</tr>
<tr>
<td>zip</td>
<td>Postal code of the shipping address.</td>
<td>No</td>
</tr>
</tbody>
</table>
You can customise the look and feel of the PayPal payment page in the following two ways:

- Using HTML Variables for Payment Page Look and Feel.

**NOTE:** HTML variables will override the account settings you save on your profile page.

This chapter also includes information on The Mobile Optimized Payment Flow.

**NOTE:** Your payment page will appear to customers after you complete your HTML integration.
Modifying Your PayPal Account Settings

The settings and appearance of your PayPal payment page can be customised. To modify your payment page, login to your account on PayPal.com. On the Profile page, under the Website payment settings section, select:

- **Settings**: to configure your payment page settings.
- **Customisation**: to specify your payment page layout and appearance.
From the **Settings** overview page you can navigate to each of the following tabs to configure the settings of your PayPal payment page:

- **Customer Information**: Specify if you would like to display the customer’s billing or shipping information on your payment page and which fields are required or editable.

- **Payment Confirmation Page**: Choose the settings for your payment confirmation page. This is the page that your customers see after a successful transaction.

- **Email Receipt**: Select to send your customers an email receipt and customise this email receipt.

- **URLs and Data Transfer**: Specify the significant URLs on your website. PayPal will redirect your customers from your PayPal payment page to the URLs you provide.

- **General Settings**: Customise the payment page button label.

The **Settings** overview page also includes a link to the [PayPal developer portal](https://developer.paypal.com) and a link to the **Customisation** page where you can modify the appearance of your payment page.

### Customer Information

From the **Customer Information** settings tab, you can select if you would like to display **Billing information** and **Shipping information** fields on your PayPal payment page. When you check the checkbox next to either **Billing information** or **Shipping information**, a list of applicable fields appears on the page. All of the listed fields will be displayed on your payment page.
From this list of customer information fields you can select which fields you would like to make editable or required on your payment page. Check the box next to each required field in the make these fields required section and next to the editable fields in the make these fields editable section.

PayPal recommends that you display as few options as possible on the payment page. Also, if you choose to collect information on the payment page that you have already collected on your website, this creates a poor buyer experience. For example, you may have already collected shipping address on your own website to calculate the shipping cost. Therefore, it is not necessary to collect this information again on the payment page.
Instead of collecting the buyer’s shipping address again on the PayPal payment page, you can pass to PayPal the shipping address information which you already collected. PayPal will then display the pre-filled and editable shipping address fields on your payment page. For more information on passing values to the PayPal payment page go to: “HTML Variables for Payment Page Settings” on page 17.

From the Customer Information page tab you can choose whether to display the following customer billing information or shipping information fields on your payment page. You can also determine which of these fields will be required and/or editable.

**NOTE:** Some merchants are required to pass the customer’s billing information with every transaction. It is recommended that you test your integration first, particularly if you plan to use the iFrame integration, to determine if the billing information fields are required.

<table>
<thead>
<tr>
<th>Billing Information Fields</th>
<th>Shipping Information Fields</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td><strong>Name</strong></td>
</tr>
<tr>
<td>• Company Name</td>
<td>• Company Name</td>
</tr>
<tr>
<td>• First Name</td>
<td>• First Name</td>
</tr>
<tr>
<td>• Last Name</td>
<td>• Last Name</td>
</tr>
<tr>
<td><strong>Address</strong></td>
<td><strong>Address</strong></td>
</tr>
<tr>
<td>• Address 1</td>
<td>• Address 1</td>
</tr>
<tr>
<td>• Address 2</td>
<td>• Address 2</td>
</tr>
<tr>
<td>• City</td>
<td>• City</td>
</tr>
<tr>
<td>• State/Province/Region</td>
<td>• State/Province/Region</td>
</tr>
<tr>
<td>• ZIP/Postal code</td>
<td>• ZIP/Postal code</td>
</tr>
<tr>
<td>• Country</td>
<td>• Country</td>
</tr>
<tr>
<td><strong>Contact Information</strong></td>
<td><strong>Contact Information</strong></td>
</tr>
<tr>
<td>• Phone number</td>
<td>• Phone number</td>
</tr>
<tr>
<td>• Email address</td>
<td>• Email address</td>
</tr>
</tbody>
</table>

You can also override the Customer Information tab settings at transaction time by passing HTML variables to the PayPal payment page. For more information see: “HTML Variables for Payment Page Look and Feel” on page 38.

**Payment Confirmation Page**

From the Payment Confirmation Page tab, you can specify which page the buyer will see after making a successful payment. You can choose to display either a PayPal confirmation page or a payment confirmation page hosted on your website:
Customising Your PayPal Payment Page

Modifying Your PayPal Account Settings

**FIGURE 3.3 Settings - Payment Confirmation Page**

- **On my site’s confirmation page**
  
  If you select this option, you are asking PayPal to redirect the buyer to your own payment confirmation page hosted on your website. You will be asked to provide the *Return URL*. The Return URL is the first page the customer sees on your site after leaving the PayPal pages. If you choose to host your own payment confirmation page, enter the path to your payment confirmation page in the Return URL field.
  
  When PayPal redirects the buyer to a payment confirmation page hosted on your website, you will have to include a message on this page about the outcome of the transaction. Be sure to display the specifics of the order using the Transaction ID that is passed back to you.

- **On a PayPal confirmation page**
  
  By selecting this option you are choosing to display a PayPal hosted confirmation page to the buyer. You can customise your PayPal hosted confirmation page with the following information:
  - (Optional) header text
– (Optional) footer text
– Customer service phone number.
  To update your customer service contact information go back to the Profile page.
– Customer service email address.
  To update your customer service contact information go back to the Profile page.
– Return URL
  The Return URL is the first page the customer sees on your site after leaving the PayPal pages. In the Return URL field, enter the path to a page on your site that you would like buyers to see after the PayPal payment confirmation page. The buyer returns to your site by clicking a link on the PayPal confirmation page. You can customise the link text.
– Text for return button / link
  Here you can specify the text of the link on the PayPal confirmation page that sends the user back to your site.
  To change how information is sent from PayPal to your web server, via HTTP POST or GET, go to the URLs and Data Transfer settings tab.

Email Receipt

From the Email Receipt settings tab, you can specify if you would like PayPal to send an email receipt to the buyer after a successful transaction. This feature is disabled by default. To enable it, check the checkbox next to Allow PayPal to email receipts.

You can customise the following information in the email receipt page:

● Sender’s email address
● Reply-to email address (optional)
● Display your business name or logo?
  You can select from the following options:
  – Show my business name as text
  – Show my business logo
  – Do not display
● Header text (optional)
● Footer text (optional)
● Customer service phone number
● Customer service email address

To update your customer service contact information, go back to the Profile page.
URLs and Data Transfer

The URLs and Data Transfer settings help you manage how data is transferred between PayPal and your website (via GET or POST). You can also set your Cancel and Error URL values here. To modify the Return URL, go to the Payment Confirmation Page.
You can configure the following settings from the *URLs and Data Transfer* tab:

**Data transfer method**
Select the HTTP method you would like PayPal to use to send back payment data to your web server. This data transfer method applies to Return, Cancel and Error URLs. The options are:

- GET
- POST

**Cancel URL**
If a customer chooses to cancel a payment, this link will direct the customer back to a page on your site that you specify.
Customising Your PayPal Payment Page

Modifying Your PayPal Account Settings

- **Cancel URL**
  Enter the URL of the page on your website where the buyer should return to if the buyer chooses to cancel the payment.
  You can also set the Cancel URL dynamically at transaction time by passing a HTML variable to PayPal. This HTML variable will override the Cancel URL specified in your account settings. For more information, go to: “HTML Variables for Payment Page Settings” on page 17.

- **Text for Cancel URL**
  Here you can customize the text of the payment cancellation link displayed on your PayPal payment page.

**Error URL**
This is the URL of the page on your website that PayPal should redirect buyers to if an error occurs during the payment transaction.

**General Settings**
The General Settings page helps you customise PayPal payment page further.

**Payment page button text**
You can customise the text on the Pay Now button.

*Figure 3.6 Settings - General Settings*
Customisation

The Customisation page allows you to customise the layout and appearance of your PayPal payment page. Here you can select one of the available page layouts. Depending on which layout you select, you can customise the header, the background color, the subheader text, the color of the pay button, the color of the button text, and the order summary section of your payment page. PayPal payment pages are optimized for both desktop and mobile browsers. To learn more about payment page behavior with mobile browsers see “The Mobile Optimized Payment Flow” on page 36.

At the top of the page, you will see the four layout templates that PayPal offers. Template A is the default template. You can select any one of these templates. If you would like an iFrame integration, select the MiniLayout template.

After you select the layout template you prefer, you can customise individual sections of this template. To make changes to a section, left-click on the section you are trying to modify or on the corresponding Click to Edit button for that section. A pop-up window will appear containing the settings applicable to that section, such as a color selector to change the color or a field for you to enter text.

When you have completed customising your layout template, click one of the following buttons:

- **Preview** - Preview the changes you have made to your template before saving and publishing it.
- **Save and Publish** - Save all of the changes you have made and publish the updated template. Your buyers will see the updated payment page.
- **Cancel** - Discard all the changes you have made in this session.
- **Previous Version** - Discard all changes you have made since the last time you saved the template. Your buyers will see the last saved template.

**Note:** You must make all modifications (including changing templates) within the same session, otherwise all changes will be lost and you will have to redo your changes. If the session times out, the design of the template will remain at the version that was last saved and published.

Choosing a Template

Choose from the following four design templates:

- **Layout A**
- **Layout B**
- **Layout C**
- **MiniLayout**
**Layout A.** Layout A is selected by default. Layout A consists of a header at the top of the page, a payment method section on the left and an order summary section on the right. You can only customise the header in this layout. You can specify the height of the header, show your company name in text or upload your company logo. You can also choose the horizontal alignment of the header.

**Note:** For Layout templates A, B and C, when the payment page is viewed from a supported mobile browser, PayPal automatically displays a mobile-optimized payment flow instead of the layout template. The mobile-optimized payment pages cannot be customized and will look the same for layouts A, B and C.

**Figure 3.7 Customisation - Layout A**
**Layout B.** Layout B has the most customisation options. Like Layout A, it consists of a header at the top of the page, a payment method section to the left and an order summary section to the right. In Layout B, you can also customise the header background color, the page background color or choose a page background image. In the payment method section on the left, you can choose the colors of the subheader text, the section border, the Pay button, and the Pay button text. In the order summary section on the right, you can customise the background color or use a background image.

**NOTE:** For Layout templates A, B and C, when the payment page is viewed from a supported mobile browser, PayPal automatically displays a mobile-optimized payment flow instead of the layout template. The mobile-optimized payment pages cannot be customized and will look the same for layouts A, B and C.
**Layout C.** In Layout C, the order summary section appears towards the top of the page, just below the header and is followed by the Payment method section. Similar to Layout B, Layout C allows you to customise the header, the page background and the colors on the payment method section. However, you cannot customise the order summary section.

**NOTE:** For Layout templates A, B and C, when the payment page is viewed from a supported mobile browser, PayPal automatically displays a mobile-optimized payment flow instead of the layout template. The mobile-optimized payment pages cannot be customized and will look the same for layouts A, B and C.

**Figure 3.9 Customisation - Layout C**
**MiniLayout.** Unlike the other layouts, MiniLayout shows only the payment method section. There is no header or order summary section. MiniLayout can either be displayed in a stand-alone popup window or framed on your site in an iFrame. For more information on the iFrame integration, see “Integrating iFrame” on page 42. The size of the MiniLayout is fixed at 490 x 565 pixels, with extra space to allow for error messages.

In MiniLayout you can customise the subheader text color, the border color, the Pay button color an the Pay button text color.

**NOTE:** For MiniLayout, when the payment page is viewed from a mobile browser, PayPal will not display a mobile-optimized flow. The reason is that if PayPal automatically displayed a mobile optimized embedded template within a merchant web page that may not be mobile optimized, this can create unexpected and undesirable results. To display a mobile optimized flow, pass either mobile or mobile-iframe in the template HTML variable at transaction time.

**Figure 3.10 Customisation - MiniLayout**
Layout Template Customisation Options
Consider the following when you customise your template:

- **Header** (applicable to Layouts A, B, and C) - By default, the business name from your profile is shown in the header. You can change the following:
  - Header height and color
  - Font type, size, and color
  - Swap between displaying the business name or the business logo image
  - Position of the business name or the logo
  - In Layouts B and C you can also change the header background color.

- **Page Background** (applicable to Layouts B and C) - You can change the following:
  - Background color
  - Background image URL

- **Payment Method Section** (applicable to Layouts B, C and MiniLayout) - You can change the following:
  - Section border color
  - Subheader text color
  - Pay Button color and button text color

- **Order Summary Section** (applicable to Layout B only) - You can change the following:
  - Section background color
  - Background image URL

The Mobile Optimized Payment Flow

The PayPal payment page is optimized for mobile browsers. If you are using layout templates A, B or C, when a buyer views the payment page from a supported mobile browser, PayPal automatically displays a mobile-optimized payment flow instead of the layout template.

**NOTE:** The mobile-optimized payment pages cannot be customized and will look the same for layouts A, B and C.

For template D, which is also known as MiniLayout or the iFrame integration, when the payment page is viewed from a mobile browser, PayPal will not display a mobile-optimized flow. The reason is that if PayPal automatically displayed a mobile optimized embedded template within a merchant web page that may not be mobile optimized, this can create unexpected and undesirable results. To display a mobile optimized flow, pass either `mobile` or `mobile-iframe` in the `template` HTML variable at transaction time. For more
Customising Your PayPal Payment Page

The Mobile Optimized Payment Flow

information, see the template HTML variable in “HTML Variables for Payment Page Look and Feel” on page 38.

**TABLE 3.1 The template HTML Variable**

<table>
<thead>
<tr>
<th>Value of template Variable</th>
<th>Behavior in a Web Browser</th>
<th>Behavior from a Mobile Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>templateA</td>
<td>Displays web layout.</td>
<td>Displays mobile optimized flow.</td>
</tr>
<tr>
<td>templateB</td>
<td>Displays web layout.</td>
<td>Displays mobile optimized flow.</td>
</tr>
<tr>
<td>templateC</td>
<td>Displays web layout.</td>
<td>Displays mobile optimized flow.</td>
</tr>
<tr>
<td>templateD</td>
<td>Displays iFrame web layout.</td>
<td>Displays iFrame web layout.</td>
</tr>
<tr>
<td>mobile</td>
<td>Displays default web layout.</td>
<td>Displays mobile optimized flow.</td>
</tr>
<tr>
<td>mobile-iframe</td>
<td>Displays default web layout.</td>
<td>Displays mobile iFrame layout.</td>
</tr>
</tbody>
</table>

Figure 3.11, “Mobile Payment Flow for a PayPal Payment” and Figure 3.12, “Mobile Payment Flow for a Credit Card Payment” show examples of the mobile optimized payment flow.

**FIGURE 3.11 Mobile Payment Flow for a PayPal Payment**
Customising Your PayPal Payment Page

HTML Variables for Payment Page Look and Feel

The table below lists the HTML variables that you can use to customise the look and feel of your payment page. The HTML variables that you pass at transaction time will override your payment page settings in your PayPal account profile. For additional HTML variables, refer to: HTML Variables for Payment Page Settings.

**NOTE:** The values you pass must not contain any of these special characters `{}`<>";

**Table 3.2 HTML Variables for Look and Feel of Payment Page**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bodyBgColor</td>
<td>Color of the surrounding background of the payment page.</td>
</tr>
<tr>
<td>bodyBgImg</td>
<td>Image of the surrounding background of the payment page. The file extension can be .gif, .jpg, .jpeg, or .png format.</td>
</tr>
<tr>
<td>footerTextColor</td>
<td>Color of the footer text.</td>
</tr>
<tr>
<td>headerBgColor</td>
<td>Color of the header background.</td>
</tr>
<tr>
<td>headerHeight</td>
<td>Height of the header banner. It can be from 50 to 140 pixels. The width cannot be changed. It is always 940 pixels.</td>
</tr>
<tr>
<td>logoFont</td>
<td>Font type of the logo text.</td>
</tr>
</tbody>
</table>
### Table 3.2 HTML Variables for Look and Feel of Payment Page

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>logoFontColor</td>
<td>Color of the logo text.</td>
</tr>
<tr>
<td>logoFontSize</td>
<td>Font size of the logo text.</td>
</tr>
<tr>
<td>logoImage</td>
<td>Image displayed in the logo. The acceptable file extension formats are .gif, .jpg, .jpeg, or .png. The width of the image cannot be more than 940 pixels.</td>
</tr>
<tr>
<td>logoImagePosition</td>
<td>Position of the image in the logo.</td>
</tr>
<tr>
<td>logoText</td>
<td>Business name displayed on your profile page. This field is editable and text specified here is displayed on the header if logoImage is not specified.</td>
</tr>
<tr>
<td>orderSummaryBgColor</td>
<td>Color of the Order Summary column on the right side of the payment page. You cannot change the color of the Order Summary box.</td>
</tr>
<tr>
<td>orderSummaryBgImage</td>
<td>Background image you can put in the Order Summary Column. The acceptable file extension formats are .gif, .jpg, .jpeg, or .png.</td>
</tr>
<tr>
<td>pageButtonBgColor</td>
<td>Background color of the Pay Now button.</td>
</tr>
<tr>
<td>pageButtonTextColor</td>
<td>Color of the Pay Now button. You cannot change the text of the button.</td>
</tr>
<tr>
<td>pageTitleTextColor</td>
<td>Color of the text used in the title of the page. (Text that says “Choose a way to pay.”)</td>
</tr>
<tr>
<td>sectionBorderColor</td>
<td>Background of the section that displays “Pay with my PayPal account” or “Pay with a Card”.</td>
</tr>
<tr>
<td>showCustomerName</td>
<td>Display customer name (first name and last name).</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default Value</strong>: true</td>
</tr>
<tr>
<td></td>
<td>- <strong>Allowable Value</strong>: true or false, where true = show and false = hide.</td>
</tr>
<tr>
<td>showBillingAddress</td>
<td>Display billing address information.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default Value</strong>: true</td>
</tr>
<tr>
<td></td>
<td>- <strong>Allowable Value</strong>: true or false, where true = show and false = hide.</td>
</tr>
<tr>
<td>showBillingPhone</td>
<td>Display billing phone number.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default Value</strong>: true</td>
</tr>
<tr>
<td></td>
<td>- <strong>Allowable Value</strong>: true or false, where true = show and false = hide.</td>
</tr>
<tr>
<td>showBillingEmail</td>
<td>Display email address for billing purposes.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default Value</strong>: false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Allowable Value</strong>: true or false, where true = show and false = hide.</td>
</tr>
<tr>
<td>showHostedThankyouPage</td>
<td>Display PayPal’s confirmation page.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default Value</strong>: true</td>
</tr>
<tr>
<td></td>
<td>- <strong>Allowable Value</strong>: true or false, where true = show and false = hide.</td>
</tr>
<tr>
<td>showShippingAddress</td>
<td>Display shipping address.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Default Value</strong>: false</td>
</tr>
<tr>
<td></td>
<td>- <strong>Allowable Value</strong>: true or false, where true = show and false = hide.</td>
</tr>
<tr>
<td>subheaderText</td>
<td>Color of the “Pay with my PayPal account” or “Pay with a Card” text.</td>
</tr>
</tbody>
</table>
Customising Your PayPal Payment Page

**HTML Variables for Payment Page Look and Feel**

The layout template that will be used for your payment page. You can override the layout template in your account profile settings by passing one of the following values at transaction time:

**NOTE:** For Layout templates A, B and C, when a buyer views the payment page from a supported mobile browser, PayPal automatically displays a mobile-optimized payment flow instead of the layout template. The mobile-optimized payment pages cannot be customized and will look the same for layouts A, B and C.

- **templateA** - Sets the payment page template to layout template A.
- **templateB** - Sets the payment page template to layout template B.
- **templateC** - Sets the payment page template to layout template C.
- **templateD** - Sets the payment page template to the MiniLayout (iFrame integration) template. When the payment page is viewed from a mobile browser, PayPal will not display a mobile-optimized flow. The reason is that if PayPal automatically displayed a mobile optimized embedded template within a merchant web page that may not be mobile optimized, this can create unexpected and undesirable results. To display the mobile optimized payment flow, pass either `mobile` or `mobile-iframe` in the `template` HTML variable at transaction time.

**NOTE:** If using template A, B, or C, no action is required to display mobile-optimized payment pages; PayPal auto-detects if the buyer is on a mobile browser and automatically displays a mobile-optimized flow. However, for **templateD**, which is also known as MiniLayout or the iFrame integration, to display the mobile optimized payment flow, you must pass either `mobile` or `mobile-iframe` at transaction time.

- **mobile** - Displays the mobile-optimized payment flow. Pass this value if you are using **templateD**, which is also known as MiniLayout or the iFrame integration, and you would like to display the mobile optimized flow without an iFrame.
- **mobile-iframe** - Displays a mobile-optimized page with an iFrame. Pass this value if you are using **templateD**, which is also known as MiniLayout or the iFrame integration, and you would like to display the mobile optimized flow with an iFrame.
Integrating iFrame in Your Website

PayPal offers a compact payment form that can be integrated in an iFrame on your website. Since this form is integrated on your website, the buyer never leaves your website, thereby reducing potential drop-offs. You can also maintain your checkout look and feel in the master frame that surrounds the compact payment form. The credit card fields are part of the compact payment form so you do not have to collect this information separately.

**IMPORTANT:** Due to security concerns involving iFrame, the following browsers are supported and secure to use - Internet Explorer 7.0, 8.0 and 9.0, Firefox 24, Chrome 30, Safari 4.x and 5.x. Transactions involving users of other browsers should not use the iFrame flow or abandon the transaction. Additionally, there is another risk beyond the iFrame flow - should the PayPal iFrame become the source of an attack, it would appear that the attack is originating from the merchant site; do not use the iFrame flow if you wish to avoid this additional risk.

**NOTE:** Due to spoof concerns, the form does not contain any PayPal branding.

If you would like an iFrame integration you must use the MiniLayout template. You can either select MiniLayout from the Customisation page of your PayPal account. Or you can pass the HTML variable Template=TemplateD at transaction time. The examples in this chapter use the latter HTML variable method to set the MiniLayout template.

For MiniLayout, when the payment page is viewed from a mobile browser, PayPal will NOT automatically display a mobile-optimized payment flow. The reason is that if PayPal automatically displayed a mobile optimized embedded template within a merchant web page that may not be mobile optimized, this can create unexpected and undesirable results. To display a mobile optimized flow, pass either mobile or mobile-iframe in the template HTML variable at transaction time.

The MiniLayout template (the compact payment form) contains the following fields:

- Credit card number
- Expiry date
- CVV2 number (if applicable, based on the card type)
- Any additional fields required for other card types, such as start date and issue number for Maestro or Switch.

This template also offers the following options:

- Remove the Pay with PayPal button. Although the form offers the option to pay using PayPal account by default, you can contact your account manager or customer support to turn this option off.
Integrating iFrame in Your Website

Integrating iFrame

- Manually customize the color of the Pay Now button.

**IMPORTANT:** This compact payment form does not display the buyer's billing address, even if showBillingAddress=true is passed. However, for some merchants, it may be necessary to pass the billing address to successfully process the transactions.

Integrating iFrame

Choose one of the following methods to integrate the compact payment form in your website:
- Manual Integration
- API Integration

**IMPORTANT:** For optimal performance, PayPal recommends that you load the iFrame asset before other assets such as images and JavaScripts. If there are too many assets running on your payment page when you load iFrame, the request for the iFrame may not be placed or might get delayed. This may result in the buyers seeing a blank iFrame.

Manual Integration

To manually integrate the compact payment form in your website, perform the following steps:

1. Enter the iFrame tag at the location where you want the compact payment form to appear on your website. For example:

   `<iframe name="hss_iframe" width="570px" height="540px"></iframe>`

   The allowable size for the compact payment form is 570 pixels in width to 540 pixels in height.
2. Below is the iFrame code, add the hidden form that is populated with the appropriate Website Payments Plus variables (including the total amount to be paid) and specify the variable TemplateD. For example,

```html
<form style="display:none" target="hss_iframe" name="form_iframe" method="post"
  <input type="hidden" name="cmd" value="_hosted-payment">
  <input type="hidden" name="subtotal" value="50">
  <input type="hidden" name="business" value="HNZ3QZMCPBAAA">
  <input type="hidden" name="paymentaction" value="sale">
  <input type="hidden" name="template" value="templateD">
  <input type="hidden" name="return" value="https://yourwebsite.com/receipt_page.html">
</form>

**NOTE:** If the iFrame transactions fail, pass the billing address. For more a table of HTML variables, see HTML Variables for Payment Page Settings.

3. Ensure that the target name matches the iFrame name, as highlighted in the example below:

```html
<iframe name="hss_iframe" width="570px" height="540px"></iframe>
<form style="display:none" target="hss_iframe" name="form_iframe" method="post"
  <input type="hidden" name="cmd" value="_hosted-payment">
  <input type="hidden" name="subtotal" value="50">
  <input type="hidden" name="business" value="HNZ3QZMCPBAAA">
  <input type="hidden" name="paymentaction" value="sale">
  <input type="hidden" name="template" value="templateD">
  <input type="hidden" name="return" value="https://yourwebsite.com/receipt_page.html">
</form>
```

4. Submit the form using JavaScript. For example:

```html
<script type="text/javascript">
  document.form_iframe.submit();
</script>
```

**Example of Manual Integration**

The complete example as per the above steps is as follows:

```html
<iframe name="hss_iframe" width="570px" height="540px"></iframe>
<form style="display:none" target="hss_iframe" name="form_iframe" method="post"
  <input type="hidden" name="cmd" value="_hosted-payment">
  <input type="hidden" name="subtotal" value="50">
  <input type="hidden" name="business" value="HNZ3QZMCPBAAA">
  <input type="hidden" name="paymentaction" value="sale">
  <input type="hidden" name="template" value="templateD">
  <input type="hidden" name="return" value="https://yourwebsite.com/receipt_page.html">
</form>
```
Integrating iFrame in Your Website

Integrating iFrame

```javascript
<script type="text/javascript">
    document.form_iframe.submit();
</script>
```

API Integration

To integrate the compact payment form in your website using the API, refer to “Using the Button Manager API with Website Payments Plus Checkout” on page 48.

**Note:** Use template=templateD for this type of integration.

There are two options in the response to initiate the Website Payments Plus payment flow:

- Using URL Returned in the Response
- Using Form POST
Integrating iFrame in Your Website

Integrating iFrame

Using URL Returned in the Response

In the URL identified in the response as EMAILLINK, add “src” for iFrame as per the example below to redirect the buyer and initiate the payment flow.

```html
<iframe src="https://securepayments.paypal.com/...?hosted_button_id=HSSS-DDrPduW-ADwkFDMjQmpUK1gTDDr.tv5alaGS61.XWVVBlMTMQEnGNoLakufqb89zTjf6" width="570px" height="540px"></iframe>
```

The allowable size of the compact payment form is 570 pixels in width to 540 pixels in height.

**IMPORTANT:** This option is not supported by Safari browsers. Use the Form POST option described below.

Using Form POST

Identify the WEBSITECODE in the response and use the code to create a **Pay Now** button on your review page. When your buyer clicks the button, they are redirected to the PayPal hosted payment page. Like the URL, the button is usable for approximately two hour or until the payment is successful.

1. Enter the iFrame tag at the location where you want the compact payment form to appear on your website. For example:

   ```html
   <iframe name="hss_iframe" width="570px" height="540px"></iframe>
   ```

   The allowable size for the compact payment form is 570 pixels in width to 540 pixels in height.

2. Insert the following in the iFrame tag:

   ```html
   WEBSITECODE=<form
   <input type="hidden" name="hosted_button_id" value="HSSS-DDrPduW-ADwkFDMjQmpUK1gTDDr.tv5alaGS61.XWVVBlMTMQEnGNoLakufqb89zTjf6">
   <input type="image" src="https://www.paypal.com/i/btn/btn_paynow_LG.gif" border="0" name="submit" alt="PayPal - The safer, easier way to pay online.">
   <img alt="" border="0" src="https://www.paypal.com/i/scr/pixel.gif" width="1" height="1">
   </form>
   ```

3. Submit the form using JavaScript. For example:

   ```html
   <script type="text/javascript">
   document.form_iframe.submit();
   </script>
   ```
Example of API (Form POST) Integration

The complete example as per the above steps is as follows:

```html
<iframe name="hss_iframe" width="570px" height="540px"></iframe>

  <input type="hidden" name="hosted_button_id" value="HSSS-GDrPDzuW-ADwkFDMjQmpUKigTDDdR.tvSalaGS61.XWVVB1MTMQEnGNoLakufQb892Tjf6"/>
  <input type="image" src="https://www.paypal.com/i/btn/btn_paynow_LG.gif" border="0" name="submit" alt="PayPal - The safer, easier way to pay online.">
  <img alt="" src="https://www.paypal.com/i/scr/pixel.gif" width="1" height="1">
</form>
<script type="text/javascript">
  document.form_iframe.submit();
</script>
```
Using the Button Manager API, you can programmatically create the checkout button that initiates the Website Payments Plus checkout flow on your website. You create the checkout button dynamically when the buyer attempts to checkout on your website. When the buyer clicks on your checkout button, you call the BMCreateButton API operation to create a checkout button and display it on your web page. When the buyer clicks on this checkout button, he or she is redirected to the PayPal payment page where he or she can complete the payment.

HTML variables control the presentation of the PayPal payment pages, the information contained on the pages, and pages to which your buyer can be returned to when leaving the payment pages. For a list of HTML variables, refer to “HTML Variables for Payment Page Settings” on page 17.

For complete information on the Button Manager API, see the Button Manager API Reference.

**NOTE:** Using the Button Manager API, you can programmatically create the checkout button, but not manage, edit, or delete it.

### Button Hosting

For Website Payments Plus, using `BUTTONTYPE=PAYMENT`, you can create the following kinds of checkout buttons:

- **Token buttons** (`BUTTONCODE=TOKEN`) are not stored on PayPal. They are dynamic and initiate the Website Payments Plus checkout flow. The parameters associated with this kind of button are secure. They have a lifetime of approximately two hours, if not used.

- **Encrypted buttons** (`BUTTONCODE=ENCRYPTED`) are not stored on PayPal. The parameters associated with this kind of button are encrypted. You must update the web pages that use an encrypted button each time the button or the associated information changes.

- **Clear text buttons** (`BUTTONCODE=CLEARTEXT`) are not stored on PayPal. The parameters associated with this kind of button are not encrypted. They can be used dynamically because you can change the value of a parameter anytime. However, if you use them, recognize the risk that sensitive information can be altered or stolen.

**NOTE:** Hosted buttons (`BUTTONCODE=HOSTED`) are not supported for Website Payments Plus and will return an error if used.

For more information about Button Manager API usage, refer to “BMCreateButton API Operation” on page 53.
Using the Button Manager API with Website Payments Plus Checkout

To create a checkout button on your website, you call the BMCreateButton API operation. When the buyer clicks on the checkout button, PayPal initiates the Website Payments Plus checkout flow.

You must decode the response from the BMCreateButton API. For information about URL encoding, refer to the Name-Value Pair API Developer Guide.

There are two options in the response to initiate the Website Payments Plus payment flow:

- Using URL Returned in the Response (Recommended)
- Using Form POST

Using URL Returned in the Response (Recommended)

Using the URL identified in the response as EMAILLINK, you can redirect the buyer and initiate the payment flow.

Example of TOKEN

EMAILLINK=https://securepayments.paypal.com/webapps/HostedSoleSolutionApp/webflow/sparta/hostedSoleSolutionProcess?hosted_button_id=HSSS-GDrPDzuW-ADwkFDmjQmpUK1gTDxR.tv5alaGS61.XWVVB1MTMQEnGNoLakufQb89zTjf6&HOSTEDBUTTONID=HSSS-GDrPDzuW-ADwkFDmjQmpUK1gTDxR.tv5alaGS61.XWVVB1MTMQEnGNoLakufQb89zTjf6&TIMESTAMP=2010-09-15T00:30:35Z&CORRELATIONID=2fc9a340a2356&ACK=Success&VERSION=65.0&BUILD=1500269
Example of ENCRYPTED


Example of CLEARTEXT


The button variables control the presentation in the flow and the information that is provided to the buyer initially. Because the URL is associated with a Website Payments Plus token, the URL is usable for approximately two hours after it is created or until the payment is successful.

Using Form POST

![Diagram of PayPal integration process]

Identify the WEBSITECODE in the response and use the code to create a Pay Now button on your review page. When your buyer clicks the button, they are redirected to the PayPal hosted
Integrating Your Website Using API

Using the Button Manager API with Website Payments Plus Checkout

Like the URL, the button is usable for approximately two hours or until the payment is successful.

Example for TOKEN

```
  <input type="hidden" name="hosted_button_id" value="HSSS-<DrPDzuW-ADwkPDMj0mUKlgTdxR.tv5alaGS61.XWVVB1MTQ6EnGNoLakuQb89zTjf6">
  <input type="image" src="https://www.paypal.com/en_US/i/btn/btn_auto_billing_LG.gif" border="0" name="submit" alt="PayPal - The safer, easier way to pay online.">
  <img alt="" border="0" src="https://www.paypal.com/en_GB/i/scr/pixel.gif" width="1" height="1">
</form>
```

Example for ENCRYPTED

```
  <input type="hidden" name="encrypted" value="BEGIN PKCS7......END PKCS7">
  <input type="image" src="https://www.paypal.com/en_US/i/btn/btn_auto_billing_LG.gif" border="0" name="submit" alt="PayPal - The safer, easier way to pay online.">
  <img alt="" border="0" src="https://www.paypal.com/en_GB/i/scr/pixel.gif" width="1" height="1">
</form>
```

Example of CLEARTEXT

```
  <input type="hidden" name="cmd" value="_hosted-payment">
  <input type="hidden" name="business" value="test1@paypal.com">
  <input type="hidden" name="tax" value="2">
  <input type="hidden" name="shipping" value="3">
  <input type="hidden" name="subtotal" value="11">
  <input type="hidden" name="paymentaction" value="sale">
  <input type="hidden" name="handling" value="4">
  <input type="image" src="https://www.paypal.com/en_US/i/btn/btn_auto_billing_LG.gif" border="0" name="submit" alt="PayPal - The safer, easier way to pay online.">
  <img alt="" border="0" src="https://www.paypal.com/en_GB/i/scr/pixel.gif" width="1" height="1">
</form>
```
NOTE: The PayPal payment page displays a button identified as Pay Now to complete the checkout flow. The Pay Now button on the PayPal page is different than the Pay Now button created by the BMCreateButton API operation on the review page. You may need to change the URL for the latter in BUTTONIMAGEURL to avoid confusion. If a URL is not specified in BUTTONIMAGEURL, then the default Pay Now is used for the button.

Encrypting Buttons Using Public and Private Key

Using Encrypted Website Payments helps secure payment buttons that you generate or write manually. Encrypted Website Payments protects the HTML button code that contains pricing information by encrypting it. HTML button code that you protect by using Encrypted Website Payments cannot be altered by malicious third parties to create fraudulent payments. For detailed information, refer to Chapter 8, “Protecting Buttons by Using Encrypted Website Payments” on page 65.

Example of Initiating Website Payments Plus Checkout

The example shows the NVP parameters using BUTTONCODE=TOKEN for creating a URL to initiate the Website Payments Plus checkout flow.

```plaintext
... 
BUTTONCODE=TOKEN
BUTTONTYPE=PAYMENT
L_BUTTONVAR0=subtotal=11
L_BUTTONVAR1=tax=2
L_BUTTONVAR2=shipping=3
L_BUTTONVAR3=handling=4
L_BUTTONVAR4=first_name=Bob
L_BUTTONVAR5=last_name=Jones
L_BUTTONVAR6=address1=Gardens
L_BUTTONVAR7=address2=99343
L_BUTTONVAR8=city=Paris
L_BUTTONVAR9=state=Alsace
L_BUTTONVAR10=zip=75002
L_BUTTONVAR11=country=FR
L_BUTTONVAR12=billing_first_name=BobBilling
L_BUTTONVAR13=billing_last_name=JonesBilling
L_BUTTONVAR14=billing_address1=5088 - 061 Craven Hill Gardens
L_BUTTONVAR15=billing_address2=Apt# 199343
L_BUTTONVAR16=billing_city=Wolverhampton
L_BUTTONVAR17=billing_state=West Midlands
L_BUTTONVAR18=billing_zip=W12 4LQ
L_BUTTONVAR19=billing_country=GB
L_BUTTONVAR20=night_phone_a=408
L_BUTTONVAR21=night_phone_b=398
L_BUTTONVAR22=night_phone_c=3065
... 
```
Integrating Your Website Using API

Using the Button Manager API with Website Payments Plus Checkout

L_BUTTONVAR23=notify_url=www.yourcompany.com
L_BUTTONVAR24=custom=custom1
L_BUTTONVAR25=cancel_return=http://www.cancel_return.com
L_BUTTONVAR26=paymentaction=sale
L_BUTTONVAR27=currency_code=GBP
L_BUTTONVAR28=invoice=123456789
L_BUTTONVAR29=lc=GB
L_BUTTONVAR30=showBillingAddress=false
L_BUTTONVAR31=showShippingAddress=false
L_BUTTONVAR32=showBillingEmail=true
L_BUTTONVAR33=showBillingPhone=true
L_BUTTONVAR34=showCustomerName=true
L_BUTTONVAR35=showCardInfo=true
L_BUTTONVAR36=showHostedThankyouPage=true
L_BUTTONVAR37=bn=GBD
L_BUTTONVAR38=cbt=Merchant ABCD
L_BUTTONVAR39=address_override=false
L_BUTTONVAR40=cpp_header_image=Red
L_BUTTONVAR41=logoText=YourCompanyLogo
L_BUTTONVAR42=logoImage=http://yourcompanyimages.com/logo.gif
L_BUTTONVAR43=logoImagePosition=center
L_BUTTONVAR44=logoFont=Arial Bold
L_BUTTONVAR45=logoFontSize=24px
L_BUTTONVAR46=logoFontColor=#f0ff0f
L_BUTTONVAR47=bodyBgImg=sdf
L_BUTTONVAR48=bodyBgColor=#AEAEAE
L_BUTTONVAR49=headerHeight=70px
L_BUTTONVAR50=headerBgColor=#4B6BA1
L_BUTTONVAR51=PageTitleTextColor=Blue
L_BUTTONVAR52=PageCollapseBgColor=Blue
L_BUTTONVAR53=PageCollapseTextColor=Yellow
L_BUTTONVAR54=PageButtonBgColor=Blue
L_BUTTONVAR55=PageButtonBgColor=Blue
L_BUTTONVAR56=orderSummaryBgColor=#EDF2F7
L_BUTTONVAR57=orderSummaryBgImage=http://teamworks/scr_bg_ordersummary.jpg
L_BUTTONVAR58=footerTextColor=#333333
L_BUTTONVAR59=footerTextlinkColor=#00f00f
L_BUTTONVAR60=template=templateC
L_BUTTONVAR61=return=http://www.yourcompany.com
BMCreateButton API Operation

Use the BMCreateButton API operation to create Website Payments Plus checkout button.

- BMCreateButton Request
- BMCreateButton Response
- BMCreateButton Errors

BMCreateButton Request

The request fields specify the characteristics of your button, which include associated menu items related to the button. You can specify up to 5 menu items, each of which can include up to 10 possible selections.

BMCreateButton Request Fields

**NOTE:** The values you pass must not contain any of these special characters (){}<>";

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>METHOD</td>
<td>(Required)</td>
</tr>
<tr>
<td>BUTTONCODE</td>
<td></td>
</tr>
<tr>
<td>BUTTONTYPE</td>
<td>(Required)</td>
</tr>
<tr>
<td>L_BUTTONVARn</td>
<td>(Optional)</td>
</tr>
</tbody>
</table>

BMCreateButton Request Fields Details

**METHOD**

- **(Required)**  
  - BMCreateButton

**BUTTONCODE**

- **(Optional)** The kind of button code to create. It is one of the following values:
  - **TOKEN** - A secure button, not stored on PayPal, used only to initiate the Website Payments Plus checkout flow; default for Pay Now button. Since version 65.2
  - **ENCRYPTED** - An encrypted button, not stored on PayPal
  - **CLEARTEXT** - An unencrypted button, not stored on PayPal

**BUTTONTYPE**

- **(Required)** The kind of button you want to create.
  - PAYMENT - Pay Now button; since version 65.2

**L_BUTTONVARn**

- **(Optional)** HTML standard button variables
**BMCreateButton Response**

The response contains an ID if the button is hosted on PayPal, and code for HTML in a website and for a link in email.

**BMCreateButton Response Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WEBSITECODE</td>
<td>HTML code for web pages</td>
</tr>
<tr>
<td>EMAILLINK</td>
<td>Code for URL for the Website Payments Plus checkout flow</td>
</tr>
<tr>
<td>HOSTEDBUTTONID</td>
<td>ID of a Website Payments Plus token</td>
</tr>
</tbody>
</table>

**BMCreateButton Errors**

Refer to the [PayPal API error codes](#) page to lookup an error code number or an error message.
Testing Your Integration in Sandbox

The PayPal Sandbox is a self-contained environment within which you can prototype and test PayPal features. The PayPal Sandbox is an almost identical copy of the live PayPal website. Its purpose is to give developers a shielded environment for testing and integration purposes, and to help avoid problems that might occur while testing PayPal integration solutions on the live site. Before moving any PayPal-based application into production, you should test the application in the Sandbox to ensure that it functions as you intend and within the guidelines and standards set forth by the PayPal Developer agreement.

For complete details on using the PayPal Sandbox, refer to the Sandbox User Guide.

Sandbox Account Credentials

Create a PayPal Sandbox Business account for the country you want to test:

- Navigate to Applications > Sandbox accounts and click the Create Account button.
- Choose the country for which you want to test your integration using the Country drop-down.
- Set Bank Verified Account to Yes.
- Complete the rest of the form and click Create Account.

**Note:** You can use any name for the account, and there is no need to tick the Login in with PayPal box.

Verify Your PayPal Sandbox Business account:

- Log in to the Sandbox test site (https://www.sandbox.paypal.com) using the email address and password from your recently created PayPal Sandbox Business account.
- Click the Unverified link on the main My Account Overview page.
- Click Add Bank Account on the Get Verified and Lift Sending Limits page.
- Fill out all the fields with fictitious information.

**Note:** Sort Code and Account Number must be unique numbers.

- Click Continue, then Add Bank Account to add the test bank account.
- Navigate to the Set-up Bank Funding page and click Continue.
- Click Submit to finish the verification process.
Upgrade to a Pro account:

Click on the arrow next to your Sandbox business account email address, and then click on the Profile link that appears below the email address (highlighted below).

This opens the Sandbox account details window. From the Profile tab of this window, select Upgrade to Pro. Then click the Enable button.
Setup Personal Sandbox Accounts:

From the developer site *Applications > Sandbox accounts* page you can create multiple *business* (merchant) and *personal* (buyer) accounts, which you can use in Sandbox test transactions. See the [Sandbox User Guide](#) for more information.

You are ready to test your Website Payments Plus integration on the Sandbox test site.
Testing Your Integration and Settings

The following sections contain information for testing your integration and modifying the look and feel of your payment page in the Sandbox environment.

- **Testing Your Integration**
- **Testing Your Settings**

**Testing Your Integration**

To test your integration in the Sandbox environment, follow the steps specified in “Simple Website Payments Plus Integration” on page 15. For testing purposes, you must make the following changes in the Form POST:

1. Change the URL to point to the Sandbox environment.

   From:

   ```html
   <input type="hidden" name="cmd" value="_hosted-payment">
   </form>
   ``

   to:

   ```html
   <input type="hidden" name="cmd" value="_hosted-payment">
   </form>
   ``

2. Change the business value to the Secure Merchant ID value that is specified at the top of the Profile page of the Sandbox Test Site.

   Accordingly, the Form POST for testing purposes will be:

   ```html
   <input type="hidden" name="cmd" value="_hosted-payment">
   <input type="hidden" name="subtotal" value="50">
   <input type="hidden" name="business" value="HNZ3QZMCPBAAA">
   <input type="hidden" name="paymentaction" value="sale">
   <input type="hidden" name="return" value="https://yourwebsite.com/receipt_page.html">
   <input type="submit" name="METHOD" value="Pay Now">
   </form>
   ```
Testing Your Settings

To change the look and feel of your payment page, modify the settings of the Customisation and Settings pages in the Profile section of the Sandbox Test Site. For complete details, refer to “Modifying Your PayPal Account Settings” on page 22.
7 Order Processing

This chapter walks you through the end of end order processing experience. It includes information about verifying the status and authenticity of the order before fulfilling it.

Verifying Transaction Status and Authenticity

When the buyer successfully completes a transaction, they are redirected either to the PayPal confirmation page or to a website you specified in the return variable or on the Settings page in the Profile section (as outlined in “Simple Website Payments Plus Integration” on page 15). When the browser is redirected to the website you specified, a Transaction ID is appended to it.

**NOTE:** To ensure that the Transaction ID is appended to the return URL, log in to your PayPal merchant account and select Profile. In the Website Payments Standard and Express Checkout section of the Profile page, select Preferences and verify that Auto Return is set to On.

**Auto Return for Website Payments**

Auto Return for Website Payments brings your buyers back to your website immediately after payment completion. Auto Return applies to PayPal Website Payments, including Buy Now, Donations, Subscriptions and Shopping Cart. Learn More

![Auto Return On/Off](image)

On the same settings page, also verify that Payment Data Transfer is set to On.

**Payment Data Transfer (optional)**

Payment Data Transfer allows you to receive notification of successful payments as they are made. The use of Payment Data Transfer depends on your system configuration and your Return URL. Please note that in order to use Payment Data Transfer, you must turn on Auto Return.

![Payment Data Transfer On/Off](image)

When you receive the redirect (URL with Transaction ID), you must verify that the order was completed successfully on PayPal before sending the items to the buyer. You can do so by checking the confirmation email sent to you by PayPal or by verifying the transaction history. You can also using one of the following methods:
Validate Instant Payment Notification (IPN)

IPN enables you to receive messages about transaction payments and activity from PayPal through asynchronous, server-to-server communication. This allows you to integrate your online payments with your order fulfillment process.

Through IPN, you receive messages for the following:

- Payments and their status (pending, complete, or denied)
- Fraud Management Filter actions
- Recurring Payment activity
- Authorisations, chargebacks, disputes, reversals, and refunds.

After a transaction has been processed, PayPal sends an IPN to the notification URL specified in your transaction using the parameter `notify_url`, or in your PayPal Profile. You must verify that the Transaction ID, transaction amount, and other order specific parameters such as Invoice ID that are sent in the IPN match with the information you have in your order processing system. For more details, refer to the Instant Payment Notification (IPN) integration guide.
Execute a GetTransactionDetails API Call

Using GetTransactionDetails API, you can obtain information about a specific transaction.

If you are integrated with PayPal APIs, you can call GetTransactionDetails with the Transaction ID that was returned in the web redirect to validate the authenticity of the order. For complete details, refer to “GetTransactionDetails API” on page 71.

Fulfilling Order

After you have verified the authenticity of the payment amount and status, you can fulfil the order by shipping the item to the buyer.
Order Processing

Fulfilling Order
Using Encrypted Website Payments helps secure payment buttons that you generate or write manually. Encrypted Website Payments protects the HTML button code that contains pricing information by encrypting it. HTML button code that you protect by using Encrypted Website Payments cannot be altered by malicious third parties to create fraudulent payments.

Encrypted Website Payments relies on standard public key encryption for protection. With public and private keys, you can dynamically generate HTML code for payment buttons and encrypt the payment details before displaying the buttons on your website. The below table illustrates the sequence of actions that occur with payment buttons protected by using Encrypted Website Payments.

**Table 8.1 How Encrypted Website Payments Works**

<table>
<thead>
<tr>
<th>Website Actions</th>
<th>Buyer Action</th>
<th>PayPal Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generate a public key for the website, upload it to PayPal, and download the PayPal public certificate to the website. <strong>NOTE:</strong> Do this action only once, when you first integrate Website Payments Plus with your website.</td>
<td></td>
<td>Check the authenticity of the data by using the website’s public key, which was previously uploaded to PayPal.</td>
</tr>
<tr>
<td>Generate HTML code for a payment button.</td>
<td>Click the published PayPal payment button.</td>
<td>Decrypt the protected button code by using the PayPal private key.</td>
</tr>
<tr>
<td>Encrypt the generated code by using the PayPal public key and then signing the encrypted code with the website’s private key.</td>
<td></td>
<td>Redirect the payer’s browser to the appropriate PayPal checkout experience, as specified in the HTML variables of the decrypted button code.</td>
</tr>
</tbody>
</table>
Public Key Encryption Used by Encrypted Website Payments

Encrypted Website Payments uses public key encryption, or asymmetric cryptography, which provides security and convenience by allowing senders and receivers of encrypted communication to exchange public keys to unlock each others messages. The fundamental aspects of public key encryption are:

- **Public keys** – Public keys are created by receivers and are given to senders before they encrypt and send information. Public certificates comprise a public key and identity information, such as the originator of the key and an expiry date. Public certificates can be signed by certificate authorities, who guarantee that public certificates and their public keys belong to the named entities.
  
  You and PayPal exchange each others’ public certificates.

- **Private keys** – Private keys are created by receivers and kept to themselves.
  
  You create a private key and keep it in your system. PayPal keeps its private key on its system.

- **The encryption process** – Senders use their private keys and receivers’ public keys to encrypt information before sending it. Receivers use their private keys and senders’ public keys to decrypt information after receiving it. This encryption process also uses digital signatures in public certificates to verify the sender of the information.
  
  You use your private key and PayPal’s public key to encrypt your HTML button code. PayPal uses it’s private key and your public key to decrypt button code after people click your payment buttons.

Setting Up Certificates Before Using Encrypted Website Payments

Do the following before you use Encrypted Website Payments to protect your payment buttons:

- **Generating Your Private Key Using OpenSSL**
- **Generating Your Public Certificate Using OpenSSL**
- **Uploading Your Public Certificate to Your PayPal Account**
- **Downloading the PayPal Public Certificate from the PayPal Website**

PayPal uses only X.509 public certificates, not public keys. A public key can be used for decryption but contains no information identifying who provided the key. A public certificate includes a public key along with information about the key, such as when the key expires and who the key belongs to. PayPal accepts public certificates in OpenSSL PEM format from any established certificate authority, such as VeriSign.

You can generate your own private key and public certificate using open source software such as OpenSSL ([https://www.openssl.org](https://www.openssl.org)), which is detailed in the following section.
Generating Your Private Key Using OpenSSL

Using the openssl program, enter the following command to generate your private key. The command generates a 1024-bit RSA private key that is stored in the file my-prvkey.pem:

```
openssl genrsa -out my-prvkey.pem 1024
```

Generating Your Public Certificate Using OpenSSL

The public certificate must be in PEM format. To generate your certificate, enter the following openssl command, which generates a public certificate in the file my-pubcert.pem:

```
openssl req -new -key my-prvkey.pem -x509 -days 365 -out my-pubcert.pem
```

Uploading Your Public Certificate to Your PayPal Account

To upload your public certificate to your PayPal account:

1. Log in to your PayPal account at https://www.paypal.jp
2. Click the Profile subtab.
3. In the Hosted Payment Settings column, click Website Payment Certificates.
   The Website Payment Certificates page appears.
4. Scroll down the page to the Your Public Certificates section, and click the Add button.
   The Add Certificate page appears.
5. Click Browse, and select the public certificate that you want to upload to PayPal from your local computer.
   **Note:** The file you upload must be in PEM format.
6. Click Add.
   After your public certificate is uploaded successfully, it appears in the Your Public Certificates section of the Website Payment Certificates page.
7. Store the certificate ID that PayPal assigned to your public certificate in a secure place. You need the certificate ID that PayPal assigned to encrypt your payment buttons by using the Encrypted Website Payments software provided by PayPal.
Protecting Buttons by Using Encrypted Website Payments
Setting Up Certificates Before Using Encrypted Website Payments

Downloading the PayPal Public Certificate from the PayPal Website

To download the PayPal public certificate:

1. Log in to your PayPal account at https://www.paypal.jp
2. Click the Profile subtab.
3. In the Seller Preferences column, click the Encrypted Payment Settings link.
4. Scroll down the page to the PayPal Public Certificate section.
5. Click Download, and save the file in a secure location on your local computer.

Removing Your Public Certificate

**IMPORTANT:** If you remove your public certificate, its associated certificate ID is no longer valid for encrypting buttons, and any buttons that you generated or wrote manually for your website that use the ID will not function correctly.

To remove one or more of your public certificates:

1. Log in to your PayPal account at https://www.paypal.jp
2. Click the Profile subtab.
3. In the Seller Preferences column, click the Encrypted Payment Settings link.
4. Scroll down the page to the PayPal Public Certificate section.
5. Select the radio button next to the certificate you want to remove, and click Remove.

The Remove Certificate page appears.

6. Click Remove to confirm the removal of the public certificate that you selected.
Using Encrypted Website Payments to Protect Your Payment Buttons

Encrypted Website Payments includes Java and Microsoft Windows software to protect the payment buttons that you generate or write manually. Download the software from the following location after logging in to PayPal: https://www.paypal.com/us/cgi-bin/webscr?cmd=p/xcl/rec/ewp-code

After you download and extract the software, copy your private key, public certificate, p12 file and the PayPal public certificate to the folder where the software is located.

1. Prepare an input file of Website Payments Plus variables and values for each encrypted button that you want to generate. Each variable and value must be on a separate line, as in the following example.

   **NOTE:** The cert_id variable identifies the public certificate you uploaded to PayPal website.
   
   ```
   cert_id=Z24MFU6DSHBXQ
   cmd=_xclick
   business=sales@company.com
   item_number=1234
   custom=sc-id-789
   amount=500.00
   currency_code=USD
   tax=41.25
   shipping=20.00
   address_override=true
   address1=123 Main St
   city=Austin
   state=TX
   zip=94085
   country=US
   no_note=1
   cancel_return=http://www.company.com/cancel.htm
   ```

2. Run the encryption software using the appropriate syntax, as shown below:

   - **Java:** PPEncrypt CertFile PrivKeyFile PPCertFile InputFile OutputFile [Sandbox]
   - **Microsoft Windows:** java ButtonEncryption CertFile PKCS12File PPCertFile Password InputFile OutputFile [Sandbox]
where the argument is:

- **CertFile**: The pathname to your own public certificate
- **PKCS12File**: The pathname to the PKCS12-format of your own public certificate
- **PPCertFile**: The pathname to a copy of the PayPal public certificate
- **Password**: The passphrase to the PKCS12-format of your own public certificate
- **InputFile**: The pathname to file containing the non-encrypted Website Payments HTML Form variables
- **OutputFile**: A file name for the encrypted output
- **[Sandbox]**: The optional word Sandbox that lets you test payment buttons in the PayPal Sandbox that you protected with Encrypted Website Payments

3. Copy the encrypted code to your website.

**Blocking Unprotected and Non-encrypted Website Payments**

For extra security of your protected and encrypted buttons, update your PayPal account profile to block unprotected and non-encrypted payments.

To block payments from unprotected and non-encrypted Website Payments Standard buttons:

1. Log in to your PayPal account at [https://www.paypal.jp](https://www.paypal.jp)
2. Click the Profile subtab.
3. In the Seller Preferences column, click the Encrypted Payment Settings link.
4. Scroll down to the Encrypted Website Payments section.
5. Next to the Block Non-encrypted Website Payment label, select the On radio button.
6. Scroll to the bottom of the page, and click Save.
Optional API Operations

Using the Transaction ID, a number of PayPal API operations are available. The most commonly used for Website Payments Plus are:

- GetTransactionDetails API
- RefundTransaction API
- DoCapture API

GetTransactionDetails API

Obtain information about a specific transaction.

- GetTransactionDetails Request
- GetTransactionDetails Response

GetTransactionDetails Request

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>METHOD</td>
<td>Must be GetTransactionDetails.</td>
</tr>
<tr>
<td>TRANSACTIONID</td>
<td>(Required) Unique identifier of a transaction.</td>
</tr>
</tbody>
</table>

**NOTE:** The details for some kinds of transactions cannot be retrieved with GetTransactionDetails. You cannot obtain details of bank transfer withdrawals, for example.

**Character length and limitations** - 17 single-byte alphanumeric characters.
Optional API Operations
GetTransactionDetails API

GetTransactionDetails Response

**NOTE:** All fields defined in the formal structure of GetTransactionDetails Response are not necessarily returned. Data is returned in a response only if PayPal has recorded data that corresponds to the field.

- Receiver Information Fields
- Payer Information Fields
- Payer Name Fields
- Address Fields
- Payment Information Fields
- Payment Item Information Fields
- Payment Item Fields
- Auction Fields
- Subscription Term Fields

**Receiver Information Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| RECEIVERMAIL | Primary email address of the payment recipient (the seller).  
If you are the recipient of the payment and the payment is sent to your non-primary email address, the value of Receiver is still your primary email address.  
**Character length and limitations** - 127 single-byte alphanumeric characters |
| RECEIVERID  | Unique account ID of the payment recipient (the seller).  
This value is the same as the value of the recipient's referral ID. |
### Payer Information Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>EMAIL</strong></td>
<td>Email address of payer.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 127 single-byte characters</td>
</tr>
<tr>
<td><strong>PAYERID</strong></td>
<td>Unique PayPal customer account identification number.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 13 single-byte alphanumeric characters</td>
</tr>
<tr>
<td><strong>PAYERSTATUS</strong></td>
<td>Status of payer. Valid values are:</td>
</tr>
<tr>
<td></td>
<td>• verified</td>
</tr>
<tr>
<td></td>
<td>• unverified</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 10 single-byte alphabetic characters</td>
</tr>
<tr>
<td><strong>SHIPTOCOUNTRYCODE</strong></td>
<td>Payer’s country of residence in the form of ISO standard 3166 two-character country codes.</td>
</tr>
<tr>
<td><strong>PAYERBUSINESS</strong></td>
<td>Payer’s business name.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 127 single-byte characters</td>
</tr>
</tbody>
</table>

### Payer Name Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SALUTATION</strong></td>
<td>Payer’s salutation.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 20 single-byte characters</td>
</tr>
<tr>
<td><strong>FIRSTNAME</strong></td>
<td>Payer’s first name.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 25 single-byte alphanumeric characters</td>
</tr>
<tr>
<td><strong>MIDDLENAME</strong></td>
<td>Payer’s middle name.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 25 single-byte alphanumeric characters</td>
</tr>
<tr>
<td><strong>LASTNAME</strong></td>
<td>Payer’s last name.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 25 single-byte alphanumeric characters</td>
</tr>
</tbody>
</table>
### Table A.4  Payer Name Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUFFIX</td>
<td>Payer’s suffix.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 12 single-byte alphanumeric characters.</td>
</tr>
</tbody>
</table>

### Address Fields

### Table A.5  Address Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDRESSOWNER</td>
<td>eBay company that maintains this address.</td>
</tr>
<tr>
<td></td>
<td><strong>Valid values are:</strong></td>
</tr>
<tr>
<td></td>
<td>● eBay</td>
</tr>
<tr>
<td></td>
<td>● PayPal</td>
</tr>
<tr>
<td>ADDRESSSTATUS</td>
<td>Status of street address on file with PayPal.</td>
</tr>
<tr>
<td></td>
<td><strong>Valid values are:</strong></td>
</tr>
<tr>
<td></td>
<td>● none</td>
</tr>
<tr>
<td></td>
<td>● Confirmed</td>
</tr>
<tr>
<td></td>
<td>● Unconfirmed</td>
</tr>
<tr>
<td>SHIPTONAME</td>
<td>Person’s name associated with this address</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 32 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOSTREET</td>
<td>First street address</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 100 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOSTREET2</td>
<td>Second street address</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 100 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOCITY</td>
<td>Name of the city</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 100 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOSTATE</td>
<td>State or province</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 40 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOZIP</td>
<td>US zip code or other country-specific postal code.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 40 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOCOUNTRYCODE</td>
<td>Expanded name of country</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 20 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOPHONENUMBER</td>
<td>Country code.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - two single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOPHONENUM</td>
<td>Country code.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - Two single-byte characters.</td>
</tr>
</tbody>
</table>
# Optional API Operations

## GetTransactionDetails API

### Payment Information Fields

**TABLE A.6  Payment Information Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRANSACTIONID</td>
<td>Unique transaction ID of the payment. Character length and limitations: 17 single-byte characters</td>
</tr>
</tbody>
</table>
| PARENTTRANSACTIONID| Parent or related transaction identification number. This field is populated for the following transaction types:  
  - Reversal - Capture of an authorised transaction.  
  - Reversal - Reauthorisation of a transaction.  
  - Capture of an order - The value of ParentTransactionID is the original OrderID.  
  - Authorisation of an order - The value of ParentTransactionID is the original OrderID.  
  - Capture of an order authorisation.  
  - Void of an order - The value of ParentTransactionID is the original OrderID.  
  **Character length and limitations** - 16 digits in xxxx-xxxx-xxxx-xxxx format |
| RECEIPTID          | Receipt identification number  
  **Character length and limitations** - 16 digits in xxxx-xxxx-xxxx-xxxx format |
| TRANSACTIONTYPE    | The type of transaction  
  **Valid values**  
  - cart  
  - express-checkout  
  **Character length and limitations** - 15 single-byte characters |
| PAYMENTTYPE        | Indicates whether the payment is instant or delayed.  
  **Character length and limitations** - Seven single-byte characters  
  **Valid values**  
  - none  
  - echeck  
  - instant |
| ORDERTIME          | Time/date stamp of payment. For example: 2006-08-15T17:23:15Z. |
| AMT                | The final amount charged, including any shipping and taxes from your Merchant Profile.  
  **Character length and limitations** - Does not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator is a period (.), and the optional thousands separator is a comma (,). Equivalent to nine characters maximum for USD. |
| CURRENCYCODE       | A three-character currency code. |
### Table A.6 Payment Information Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FEEAMT</td>
<td>PayPal fee amount charged for the transaction</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - Does not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator is a period (.), and the optional thousands separator is a comma (,). Equivalent to nine characters maximum for USD.</td>
</tr>
<tr>
<td>SETTLEAMT</td>
<td>Amount deposited in your PayPal account after a currency conversion.</td>
</tr>
<tr>
<td>TAXAMT</td>
<td>Tax charged on the transaction.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - Does not exceed $10,000 USD in any currency. No currency symbol. Regardless of currency, decimal separator is a period (.), and the optional thousands separator is a comma (,). Equivalent to nine characters maximum for USD.</td>
</tr>
<tr>
<td>EXCHANGERATE</td>
<td>Exchange rate if a currency conversion occurred. Relevant only if your are billing in their non-primary currency. If the customer chooses to pay with a currency other than the non-primary currency, the conversion occurs in the customer’s account.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - a decimal that does not exceed 17 characters, including decimal point.</td>
</tr>
</tbody>
</table>
### Optional API Operations

#### GetTransactionDetails API

**PAYMENTSTATUS**

Status of the payment.

The status of the payment:

- **None** - No status
- **Canceled-Reversal** - A reversal has been canceled; for example, when you win a dispute and the funds for the reversal have been returned to you.
- **Completed** - The payment has been completed, and the funds have been added successfully to your account balance.
- **Denied** - You denied the payment. This happens only if the payment was previously pending because of possible reasons described for the PendingReason element.
- **Expired** - the authorisation period for this payment has been reached.
- **Failed** - The payment has failed. This happens only if the payment was made from your customer’s bank account.
- **In-Progress** - The transaction has not terminated, e.g. an authorisation may be awaiting completion.
- **Partially-Refunded** - The payment has been partially refunded.
- **Pending** - The payment is pending. See the PendingReason field for more information.
- **Refunded** - You refunded the payment.
- **Reversed** - A payment was reversed due to a chargeback or other type of reversal. The funds have been removed from your account balance and returned to the buyer. The reason for the reversal is specified in the ReasonCode element.
- **Processed** - A payment has been accepted.
- **Voided** - An authorisation for this transaction has been voided.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAYMENTSTATUS</td>
<td>Status of the payment.</td>
</tr>
<tr>
<td></td>
<td>The status of the payment:</td>
</tr>
<tr>
<td></td>
<td>- None - No status</td>
</tr>
<tr>
<td></td>
<td>- Canceled-Reversal - A reversal has been canceled; for example, when you win a dispute and the funds for the reversal have been returned to you.</td>
</tr>
<tr>
<td></td>
<td>- Completed - The payment has been completed, and the funds have been added successfully to your account balance.</td>
</tr>
<tr>
<td></td>
<td>- Denied - You denied the payment. This happens only if the payment was previously pending because of possible reasons described for the PendingReason element.</td>
</tr>
<tr>
<td></td>
<td>- Expired - the authorisation period for this payment has been reached.</td>
</tr>
<tr>
<td></td>
<td>- Failed - The payment has failed. This happens only if the payment was made from your customer’s bank account.</td>
</tr>
<tr>
<td></td>
<td>- In-Progress - The transaction has not terminated, e.g. an authorisation may be awaiting completion.</td>
</tr>
<tr>
<td></td>
<td>- Partially-Refunded - The payment has been partially refunded.</td>
</tr>
<tr>
<td></td>
<td>- Pending - The payment is pending. See the PendingReason field for more information.</td>
</tr>
<tr>
<td></td>
<td>- Refunded - You refunded the payment.</td>
</tr>
<tr>
<td></td>
<td>- Reversed - A payment was reversed due to a chargeback or other type of reversal. The funds have been removed from your account balance and returned to the buyer. The reason for the reversal is specified in the ReasonCode element.</td>
</tr>
<tr>
<td></td>
<td>- Processed - A payment has been accepted.</td>
</tr>
<tr>
<td></td>
<td>- Voided - An authorisation for this transaction has been voided.</td>
</tr>
</tbody>
</table>
TABLE A.6 Payment Information Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PENDINGREASON</td>
<td><strong>NOTE:</strong> PendingReason is returned in the response only if PaymentStatus is Pending. The reason the payment is pending:</td>
</tr>
<tr>
<td></td>
<td>• none - No pending reason.</td>
</tr>
<tr>
<td></td>
<td>• address - The payment is pending because your customer did not include a confirmed shipping address and your Payment Receiving Preferences is set such that you want to manually accept or deny each of these payments. To change your preference, go to the Preferences section of your Profile.</td>
</tr>
<tr>
<td></td>
<td>• authorisation - The payment is pending because it has been authorised but not settled. You must capture the funds first.</td>
</tr>
<tr>
<td></td>
<td>• echeck - The payment is pending because it was made by an eCheck that has not yet cleared.</td>
</tr>
<tr>
<td></td>
<td>• intl - The payment is pending because you hold a non-U.S. account and do not have a withdrawal mechanism. You must manually accept or deny this payment from your Account Overview.</td>
</tr>
<tr>
<td></td>
<td>• multi-currency - You do not have a balance in the currency sent, and you do not have your Payment Receiving Preferences set to automatically convert and accept this payment. You must manually accept or deny this payment.</td>
</tr>
<tr>
<td></td>
<td>• order - The payment is pending because it is part of an order that has been authorised but not settled.</td>
</tr>
<tr>
<td></td>
<td>• paymentreview - The payment is pending while it is being reviewed by PayPal for risk.</td>
</tr>
<tr>
<td></td>
<td>• unilateral - The payment is pending because it was made to an email address that is not yet registered or confirmed.</td>
</tr>
<tr>
<td></td>
<td>• verify - The payment is pending because you are not yet verified. You must verify your account before you can accept this payment.</td>
</tr>
<tr>
<td></td>
<td>• other - The payment is pending for a reason other than those listed above. For more information, contact PayPal customer service.</td>
</tr>
<tr>
<td>REASONCODE</td>
<td>The reason for a reversal if TransactionType is reversal:</td>
</tr>
<tr>
<td></td>
<td>• none - No reason code</td>
</tr>
<tr>
<td></td>
<td>• chargeback - A reversal has occurred on this transaction due to a chargeback by your customer.</td>
</tr>
<tr>
<td></td>
<td>• guarantee - A reversal has occurred on this transaction due to your customer triggering a money-back guarantee.</td>
</tr>
<tr>
<td></td>
<td>• buyer-complaint - A reversal has occurred on this transaction due to a complaint about the transaction from your customer.</td>
</tr>
<tr>
<td></td>
<td>• refund - A reversal has occurred on this transaction because you have given the customer a refund.</td>
</tr>
<tr>
<td></td>
<td>• other - A reversal has occurred on this transaction due to a reason not listed above.</td>
</tr>
</tbody>
</table>
Optional API Operations
GetTransactionDetails API

**TABLE A.6 Payment Information Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROTECTIONELIGIBILITY</td>
<td>Since version 64.4, the kind of seller protection in force for the transaction, which is one of the following values:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Eligible</strong> – Seller is protected by PayPal's Seller Protection Policy for Unauthorised Payments and Item Not Received</td>
</tr>
<tr>
<td></td>
<td>• <strong>ItemNotReceivedEligible</strong> – Seller is protected by PayPal's Seller Protection Policy for Item Not Received</td>
</tr>
<tr>
<td></td>
<td>• <strong>UnauthorizedPaymentEligible</strong> – Seller is protected by PayPal's Seller Protection Policy for Unauthorized Payment</td>
</tr>
<tr>
<td></td>
<td>• <strong>Ineligible</strong> – Seller is not protected under the Seller Protection Policy</td>
</tr>
</tbody>
</table>

**Payment Item Information Fields**

**TABLE A.7 Payment Item Information Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVNUM</td>
<td>Invoice number you set in the original transaction.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 127 single-byte alphanumeric characters</td>
</tr>
<tr>
<td>CUSTOM</td>
<td>Custom field you set in the original transaction.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 127 single-byte alphanumeric characters</td>
</tr>
<tr>
<td>NOTE</td>
<td>Memo entered by your customer in PayPal Website Payments note field.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 255 single-byte alphanumeric characters</td>
</tr>
<tr>
<td>SALESTAX</td>
<td>Amount of tax charged on payment.</td>
</tr>
</tbody>
</table>
Payment Item Fields

**TABLE A.8 Payment Item Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>L.DESCn</td>
<td>Description of the nth item.</td>
</tr>
<tr>
<td></td>
<td>These parameters must be ordered sequentially beginning with 0 (for example L.DESC0, L.DESC1).</td>
</tr>
<tr>
<td>L.NUMBERn</td>
<td>Item number set by you. If this was a shopping cart transaction, PayPal</td>
</tr>
<tr>
<td></td>
<td>appends the number of the item to the HTML item_number variable. For</td>
</tr>
<tr>
<td></td>
<td>example, item_number1, item_number2, and so forth.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 127 single-byte alphanumeric</td>
</tr>
<tr>
<td></td>
<td>characters.</td>
</tr>
<tr>
<td></td>
<td>These parameters must be ordered sequentially beginning with 0 (for example L.NUMBER0, L.NUMBER1).</td>
</tr>
<tr>
<td>L.QTYn</td>
<td>Quantity set by you or entered by the customer.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - no limit</td>
</tr>
<tr>
<td>L.AMTn</td>
<td>Cost of item.</td>
</tr>
<tr>
<td></td>
<td>These parameters must be ordered sequentially beginning with 0 (for example L.AMT0, L.AMT1).</td>
</tr>
<tr>
<td>L.OPTIONSNAMEn</td>
<td>PayPal option names for an item in the shopping cart; each name corresponds</td>
</tr>
<tr>
<td></td>
<td>to an option value. There can be multiple option names per item. The</td>
</tr>
<tr>
<td></td>
<td>option names are ordered sequentially beginning with 0 (for example, L.OPTIONSNAMES0, L.OPTIONSNAME1).</td>
</tr>
<tr>
<td>L.OPTIONSVALUEn</td>
<td>PayPal option values corresponding to option names of an item in the</td>
</tr>
<tr>
<td></td>
<td>shopping cart. The option names are ordered sequentially beginning with 0</td>
</tr>
<tr>
<td></td>
<td>(for example, L.OPTIONSVALUE0, L.OPTIONSVALUE1).</td>
</tr>
</tbody>
</table>

Auction Fields

**TABLE A.9 Auction Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUYERID</td>
<td>Customer’s auction ID</td>
</tr>
<tr>
<td>CLOSINGDATE</td>
<td>Auction’s close date</td>
</tr>
<tr>
<td>MULTITITEM</td>
<td>Counter used for multi-item auction payments</td>
</tr>
</tbody>
</table>
Subscription Term Fields

**TABLE A.10 Subscription Terms Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMT</td>
<td>The amount subscriber is to be charged in one payment. Character length and limitations: no limit</td>
</tr>
<tr>
<td>PERIOD</td>
<td>The period of time that the subscriber will be charged. Character length and limitations: no limit</td>
</tr>
</tbody>
</table>

RefundTransaction API

Issue a refund to the PayPal account holder associated with a transaction.

- RefundTransaction Request
- RefundTransaction Response

RefundTransaction Request

**TABLE A.11 RefundTransaction Request Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>METHOD</td>
<td>Must be RefundTransaction.</td>
</tr>
<tr>
<td>TRANSACTIONID</td>
<td>(Required) Unique identifier of a transaction. <strong>Character length and limitations</strong> - 17 single-byte alphanumeric characters.</td>
</tr>
</tbody>
</table>
| REFUNDTYPE | (Required) Type of refund you are making:  
- Other  
- Full  
- Partial |
| AMT     | Refund amount. Amount is required if RefundType is Partial. **NOTE:** If RefundType is Full, do not set the Amount. |
| NOTE    | (Optional) Custom memo about the refund. **Character length and limitations** - 255 single-byte alphanumeric characters. |
Optional API Operations

DoCapture API

RefundTransaction Response

TABLE A.12 RefundTransaction Response Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>REFUNDTRANSACTIONID</td>
<td>Unique transaction ID of the refund.</td>
</tr>
<tr>
<td></td>
<td>Character length and limitations - 17 single-byte alphanumeric characters.</td>
</tr>
<tr>
<td>FEEREFUNDAMT</td>
<td>Transaction fee refunded to original recipient of payment.</td>
</tr>
<tr>
<td>GROSSREFUNDAMT</td>
<td>Amount of money refunded to original payer.</td>
</tr>
<tr>
<td>NETREFUNDAMT</td>
<td>Amount subtracted from PayPal balance of original recipient of payment to make this refund.</td>
</tr>
</tbody>
</table>

DoCapture API

Capture an authorized payment.

- DoCapture Request
- DoCapture Response

DoCapture Request

TABLE A.13 DoCapture Request Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>METHOD</td>
<td>(Required) Must be DoCapture.</td>
</tr>
<tr>
<td>AUTHORIZATIONID</td>
<td>(Required) The authorisation identification number of the payment you want to capture. This is the transaction id returned from DoExpressCheckoutPayment or DoDirectPayment. Character length and limitations - 19 single-byte characters maximum.</td>
</tr>
<tr>
<td>AMT</td>
<td>(Required) Amount to capture. Limitations - Value is a positive number which cannot exceed $10,000 USD in any currency. No currency symbol. Must have two decimal places, decimal separator must be a period (.), and the optional thousands separator must be a comma (,).</td>
</tr>
<tr>
<td>CURRENCYCODE</td>
<td>(Optional) A three-character currency code. Default: USD.</td>
</tr>
</tbody>
</table>
**TABLE A.13 DoCapture Request Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| COMPLETETYPE | (Required) The value Complete indicates that this the last capture you intend to make. The value NotComplete indicates that you intend to make additional captures.  
**NOTE:** If Complete, any remaining amount of the original authorised transaction is automatically voided and all remaining open authorisations are voided.  
**Character length and limitations** - 12 single-byte alphanumeric characters. |
| INVNUM | (Optional) Your invoice number or other identification number that is displayed to the merchant and customer in his transaction history.  
**NOTE:** This value on DoCapture will overwrite a value previously set on DoAuthorisation.  
**NOTE:** The value is recorded only if the authorisation you are capturing is an order authorisation, not a basic authorisation.  
**Character length and limitations** - 127 single-byte alphanumeric characters |
| NOTE | (Optional) An informational note about this settlement that is displayed to the payer in email and in his transaction history.  
**Character length and limitations** - 255 single-byte characters |
Optional API Operations
DoCapture API

**Table A.13 DoCapture Request Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| SOFTDESCRIPTOR | (Optional) The soft descriptor is a per transaction description of the payment that is passed to the consumer’s credit card statement. If a value for the soft descriptor field is provided, the full descriptor displayed on the customer’s statement has the following format: 

\(<PP * | PAYPAL*><Merchant descriptor as set in the Payment Receiving Preferences><1space><soft descriptor>\)

The soft descriptor can contain only the following characters.

- Alphanumeric characters
- - (dash)
- * (asterisk)
- . (period)
- {space}

If you use any other characters (such as "",""), an error code is returned.

The soft descriptor does not include the phone number, which can be toggled between the merchant’s customer service number and PayPal’s customer service number.

The maximum length of the total soft descriptor is 22 characters. Of this, either 4 or 8 characters are used by the PayPal prefix shown in the data format. Thus, the maximum length of the soft descriptor passed in the API request is:

\[22 - \text{len}<PP * | PAYPAL *>- \text{len}<Descriptor set in Payment ReceivingPreferences> + 1\)

For example, assume the following conditions:

- The PayPal prefix toggle is set to PAYPAL* in PayPal’s admin tools.
- The merchant descriptor set in the Payment Receiving Preferences is set to EBAY.
- The soft descriptor is passed in as JanesFlowerGifts LLC.

The resulting descriptor string on the credit card would be: PAYPAL*EBAY JanesFlow
DoCapture Response

- DoCapture Response Fields
- PayerInfo Type Fields
- Ship To Address Fields
- Payer Name Fields

**NOTE:** If you use version 56.0 or later of the DoCapture API, only the authorisation ID, transaction ID, transaction type, payment date, gross amount and payment status are guaranteed to be returned. If you need the values of other fields and they are not returned, you can obtain their values later by calling GetTransactionDetails or by using the reporting mechanism.

**DoCapture Response Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTHORIZATIONID</td>
<td>The authorisation identification number you specified in the request.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 19 single-byte characters maximum.</td>
</tr>
</tbody>
</table>

**PayerInfo Type Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMAIL</td>
<td>Email address of payer.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 127 single-byte characters.</td>
</tr>
<tr>
<td>PAYERID</td>
<td>Unique PayPal customer account identification number.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 13 single-byte alphanumeric characters.</td>
</tr>
<tr>
<td>PAYERSTATUS</td>
<td>Status of payer. Valid values are:</td>
</tr>
<tr>
<td></td>
<td>- verified</td>
</tr>
<tr>
<td></td>
<td>- unverified</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 10 single-byte alphabetic characters.</td>
</tr>
<tr>
<td>COUNTRYCODE</td>
<td>Payer’s country of residence in the form of ISO standard 3166 two-character country codes.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - Two single-byte characters.</td>
</tr>
</tbody>
</table>
**Optional API Operations**

**DoCapture API**

---

**Table A.15 PayerInfo Type Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BUSINESS</td>
<td>Payer’s business name.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 127 single-byte characters.</td>
</tr>
</tbody>
</table>

**Ship To Address Fields**

**Table A.16 Ship To Address Fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDRESSSTATUS</td>
<td>Status of street address on file with PayPal.</td>
</tr>
<tr>
<td></td>
<td>Valid values are:</td>
</tr>
<tr>
<td></td>
<td>• none</td>
</tr>
<tr>
<td></td>
<td>• Confirmed</td>
</tr>
<tr>
<td></td>
<td>• Unconfirmed</td>
</tr>
<tr>
<td>SHIPTONAME</td>
<td>Person’s name associated with this address.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 32 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOSTREET</td>
<td>First street address.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 100 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOSTREET2</td>
<td>Second street address.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 100 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOCITY</td>
<td>Name of city.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 40 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOSTATE</td>
<td>State or province.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 40 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOZIP</td>
<td>Required for U.S. addresses only.</td>
</tr>
<tr>
<td></td>
<td>U.S. ZIP code or other country-specific postal code.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - 20 single-byte characters.</td>
</tr>
<tr>
<td>SHIPTOCOUNTRYCODE</td>
<td>Country code.</td>
</tr>
<tr>
<td></td>
<td><strong>Character length and limitations</strong> - Two single-byte characters.</td>
</tr>
</tbody>
</table>
## Payer Name Fields

### TABLE A.17 Payer Name Fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SALUTATION</td>
<td>Payer’s salutation.</td>
</tr>
<tr>
<td><strong>Character length and limitations</strong> - 20 single-byte characters.</td>
<td></td>
</tr>
<tr>
<td>FIRSTNAME</td>
<td>Payer’s first name.</td>
</tr>
<tr>
<td><strong>Character length and limitations</strong> - 25 single-byte characters.</td>
<td></td>
</tr>
<tr>
<td>MIDDLENAME</td>
<td>Payer’s middle name.</td>
</tr>
<tr>
<td><strong>Character length and limitations</strong> - 25 single-byte characters.</td>
<td></td>
</tr>
<tr>
<td>LASTNAME</td>
<td>Payer’s last name</td>
</tr>
<tr>
<td><strong>Character length and limitations</strong> - 25 single-byte characters.</td>
<td></td>
</tr>
<tr>
<td>SUFFIX</td>
<td>Payer’s suffix</td>
</tr>
<tr>
<td><strong>Character length and limitations</strong> - 12 single-byte characters.</td>
<td></td>
</tr>
</tbody>
</table>
Optional API Operations

DoCapture API
Moving from Website Payments Standard to Website Payments Plus

If you are moving from Website Payment Standard to Website Payments Plus, you must make the following changes:

1. In the FORM tag, change the cmd value in the action attribute
   from:

   ```html
   <form action="https://www.paypal.com/cgi-bin/webscr" method="post">
   <input type="hidden" name="cmd" value="_xclick">
   ```
   
   to:

   ```html
   <input type="hidden" name="cmd" value="_hosted-payment">
   ```

2. Change HTML variable amount to subtotal.

   **NOTE:** Website Payment Standard has separate values for amount and quantity, whereas Website Payments Plus uses subtotal only. Ensure that the subtotal includes the amount you want to charge the buyer, taking into account any applicable discount and the quantity of items.

3. If you have already customised your checkout flow, such as color and background, you will need to redo those settings for Website Payments Plus. For more information about customising the look and feel of your payment page, refer to Chapter 3, “Customising Your PayPal Payment Page” on page 21.
Sample Code Comparison

Below is a comparison of Website Payment Standard code and Website Payments Plus code.

**NOTE:** In the samples below, note that for Website Payment Standard, amount and quantity are specified as 10 and 2 respectively. For Website Payments Plus, the subtotal is 20, which is the product of the above-mentioned amount and quantity. This is highlighted in bold below.

**Sample Website Payments Standard Code**

```html
  <input type="hidden" name="cmd" value="_xclick">
  <input type="hidden" name="business" value="seller@designerfotos.com">
  <input type="hidden" name="amount" value="10">
  <input type="hidden" name="quantity" value="2">
  <input type="hidden" name="tax" value="1">
  <input type="hidden" name="currency_code" value="USD">
  <input type="hidden" name=" invoice" value="MEM32507725">
  <!-- Enable override of payer’s stored PayPal address. -->
  <input type="hidden" name="address_override" value="1">
  <!-- Set prepopulation variables to override stored address. -->
  <input type="hidden" name="first_name" value="John">
  <input type="hidden" name="last_name" value="Doe">
  <input type="hidden" name="address1" value="345 Lark Ave">
  <input type="hidden" name="city" value="San Jose">
  <input type="hidden" name="state" value="CA">
  <input type="hidden" name="zip" value="95121">
  <input type="hidden" name="country" value="US">
  <input type="hidden" name="bn" value="CC_Partner_WPS">
  <input type="hidden" name="notify_url" value="https://www.paypal.com/IPN/">
  <input type="image" name="submit" border="0" src="https://www.paypal.com/img.gif ">
</form>
```

**Sample Website Payments Plus Code**

```html
  <input type="hidden" name="cmd" value="_hosted-payment">
  <input type="hidden" name="business" value="HNZ3QZMCPBAAA">
  <input type="hidden" name="subtotal" value="20">
  <input type="hidden" name="paymentaction" value="sale">
  <input type="hidden" name="currency_code" value="USD">
  <input type="hidden" name=" invoice" value="MEM32507725">
  <input type="hidden" name="template" value="TemplateB">
  <!-- Enable override of payer’s stored PayPal address. -->
  <input type="hidden" name="address_override" value="true">
  <!-- Set prepopulation variables to override stored address. -->
```
Moving from Website Payments Standard to Website Payments Plus
Refer to the PayPal API error codes page to lookup an error code number or an error message.
The table below lists all the currencies supported by PayPal.

**Table D.1 Currency Codes**

<table>
<thead>
<tr>
<th>Currency Code</th>
<th>Country Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUD</td>
<td>Australian Dollar</td>
</tr>
<tr>
<td>CAD</td>
<td>Canadian Dollar</td>
</tr>
<tr>
<td>CHF</td>
<td>Swiss Franc</td>
</tr>
<tr>
<td>CZK</td>
<td>Czech Koruna</td>
</tr>
<tr>
<td>DKK</td>
<td>Danish Krone</td>
</tr>
<tr>
<td>EUR</td>
<td>Euro</td>
</tr>
<tr>
<td>GBP</td>
<td>British Pound Sterling</td>
</tr>
<tr>
<td>HKD</td>
<td>Hong Kong Dollar</td>
</tr>
<tr>
<td>HUF</td>
<td>Hungarian Forint</td>
</tr>
<tr>
<td>JPY</td>
<td>Japanese Yen</td>
</tr>
<tr>
<td>NOK</td>
<td>Norwegian Krone</td>
</tr>
<tr>
<td>NZD</td>
<td>New Zealand Dollar</td>
</tr>
<tr>
<td>PLN</td>
<td>Polish Zloty</td>
</tr>
<tr>
<td>SEK</td>
<td>Swedish Krona</td>
</tr>
<tr>
<td>SGD</td>
<td>Singapore Dollar</td>
</tr>
<tr>
<td>USD</td>
<td>U.S. Dollar</td>
</tr>
</tbody>
</table>
The text content is presented in a natural format as follows:

Index

**A**
- address_override 17
- address1 17
- address2 17
- API integration 47
- appearance 31

**B**
- background 36
- billing_address1 17
- billing_address2 17
- billing_city 17
- billing_country 17
- billing_first_name 17
- billing_last_name 17
- billing_state 17
- billing_zip 17
- BMCreateButton API 47
- BMCreateButton API Operation 53
- BMCreateButton Errors 54
- BMCreateButton Request 53
- BMCreateButton Request Fields 53
- BMCreateButton Response 54
- BMCreateButton Response Fields 54
- bn 17
- bodyBgColor 38
- bodyBgImg 38
- business 17
- button hosting 47
  - clear text buttons 47
  - encrypted buttons 47
  - token buttons 47
- Button Manager API 47
- BUTTONCODE 53
- BUTTONCODE=CLEARTEXT 47
- BUTTONCODE=ENCRYPTED 47
- BUTTONCODE=HOSTED 47
- BUTTONCODE=_TOKEN 47
- BUTTONTYPE 53
- BUTTONTYPE=PAYMENT 47
- buyer_email 17

**C**
- cancel_return 18
- cbt 18
- city 18
- clear text buttons 47
- CLEARTEXT 47
- country 18
- currency codes 95
- currency_code 18
- custom 18
- customising payment page 21
- customising your template 36

**D**
- DoCapture API 82

**E**
- EMAILLINK 45, 48, 54
- ENCRYPTED 47
- encrypted buttons 47
- encryption 51
- encryption process 66
- error messages 93
- Express Checkout 13

**F**
- first_name 18
- footerTextColor 38

**G**
- GetTransactionDetails API 63, 71
Index

H
handling 18
header 36
headerBgColor 38
headerHeight 38
HOSTED 47
HOSTEDBUTTONID 54
HTML integration 15
HTTP variables 17

I
iFrames 41
  API integration 44
    manual integration 42
InputFile 70
Instant Payment Notification (IPN) 62
integration
  API 47
    HTML 15
invoice 18

L
L_BUTTONVAR 53
last_name 18
lc 18
logoFont 38
logoFontColor 39
logoFontSize 39
logoImage 39
logoImagePosition 39
logoText 39

M
METHOD 53
modifying account settings 22

N
night_phone_a 18
night_phone_b 18
night_phone_c 18
notify_url 18

O
optional API operations 71
order processing 61
order summary column 36
orderSummaryBgColor 39
orderSummaryBgImage 39
OutputFile 70
overview 11

P
pageButtonBgColor 39
pageButtonTextColor 39
pageTitleTextColor 39
paymentaction 18
PCI complicance 11
PPCertFile 70
preview 31
previous version 31
private key 51
private keys 66
  generating 67
public certificate
  downloading 68
  generating 67
  removing 68
  uploading 67
public key 51
public keys 66

R
RefundTransaction API 81
return 19

S
save and publish 31
sectionBorder 39
settings 23
shipping 19
showBillingAddress 39
showBillingEmail 39
showBillingPhone 39
showCustomerName 39
showHostedThankyouPage 39
showShippingAddress  39
simple integration  15
state  19
subheaderText  39
subtotal  19

tax  19
template  40
testing your integration  55
title  36
TOKEN  47
token buttons  47
transaction status  61

WEBSITECODE  45, 49, 54

zip  19