SecurePayTech.com Hosted Payment Page

Version 1.2

securepaytech.com
SECURE PAYMENT TECHNOLOGIES

Introduction

This document is designed to help you get started in your effort to integrate your website or system(s) with SecurePayTech's on-line credit card processing facilities. In particular, this document descibes how to use the SecurePayTech.com secure hosted payment page – the simplest method for performing real-time credit card transactions through SecurePayTech.com.

Using the SecurePayTech Hosted Payment Page

Overview

The hosted payment page operates slightly differently from other methods of integrating your website with SecurePayTech.com. Instead of your website code issuing the payment instructions directly to our payment gateway, your website contains a form which passes through some required pieces of information to a page on our site, which then collects the card information, and submits it for processing. When the payment is successful, the customer can then be directed back to your website via a link URL.

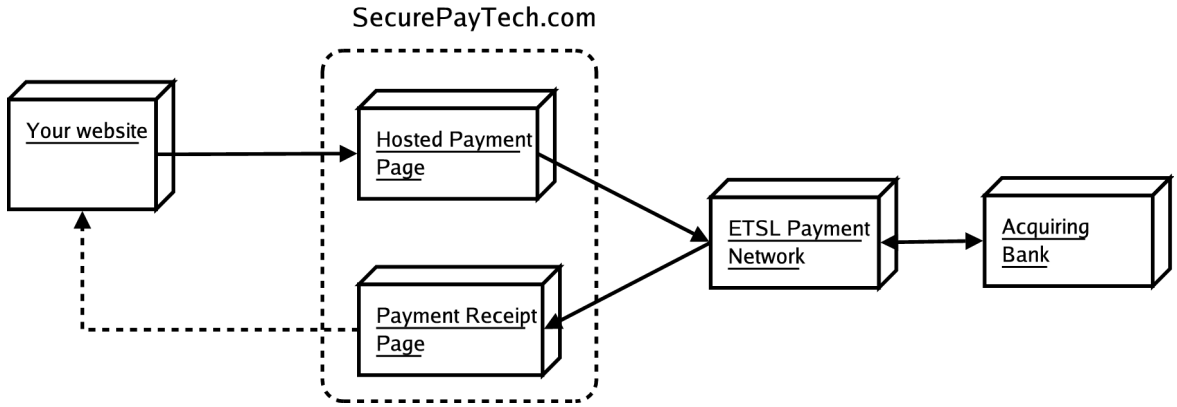

Figure 1: Overview of the Hosted Payment Page system

Payment Forms

You will need to construct a form on your website that submits the required details to our payment page.
The required details are:

- The payment dollar amount
- The order reference
- Your VPS Merchant ID number
- A link pointing to a “Return URL”. This could be a “thank-you” style page.

The optional details are:

- A flag indicating that the CSC from the card may be input
- A link pointing to a “Cancel URL”. A page to return to if a customer decides not to continue with payment.

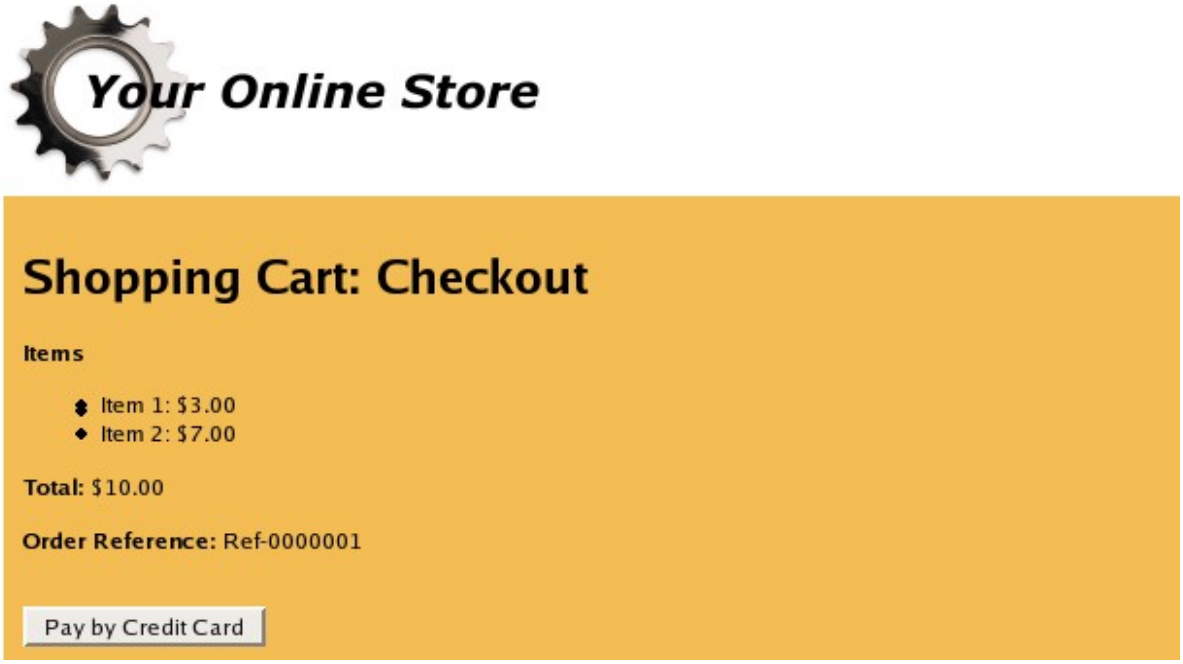

Figure 2: Example merchant website page, ready for submission to the SecurePayTech payment page

The page shown in Figure 2 above is an example of such a form. The HTML code is shown:

```
<form action="https://merchant.securepaytech.com/paymentpage/index.php"
method="post" id="mypaymentform">

 <input type="hidden" name="amount" value="10.00" />
 <input type="hidden" name="merchantID" value="TESTDIGISPL1" />
 <input type="hidden" name="returnURL"
       value="http://www.youronlinestore.co.nz/cart/thankyou.html" />
 <input type="hidden" name="cancelURL"
       value="http://www.youronlinestore.co.nz/cart/pre-payment.html" />
 <input type="hidden" name="orderReference" value="Ref-0000001"/>
 <input type="hidden" name="enableCsc"/>
 <input id="sub" type="submit"  value="Pay by Credit Card" />

</form>
```

The `enableCsc` field in the example above must be omitted unless your merchant bank supports CSC checking and your merchant account is configured with CSC checking support. If `cancelURL` is specified, a cancel button will be shown next to the submit button.

Clicking the “Pay” button on this form will send the browser to the SecurePayTech.com payment page:

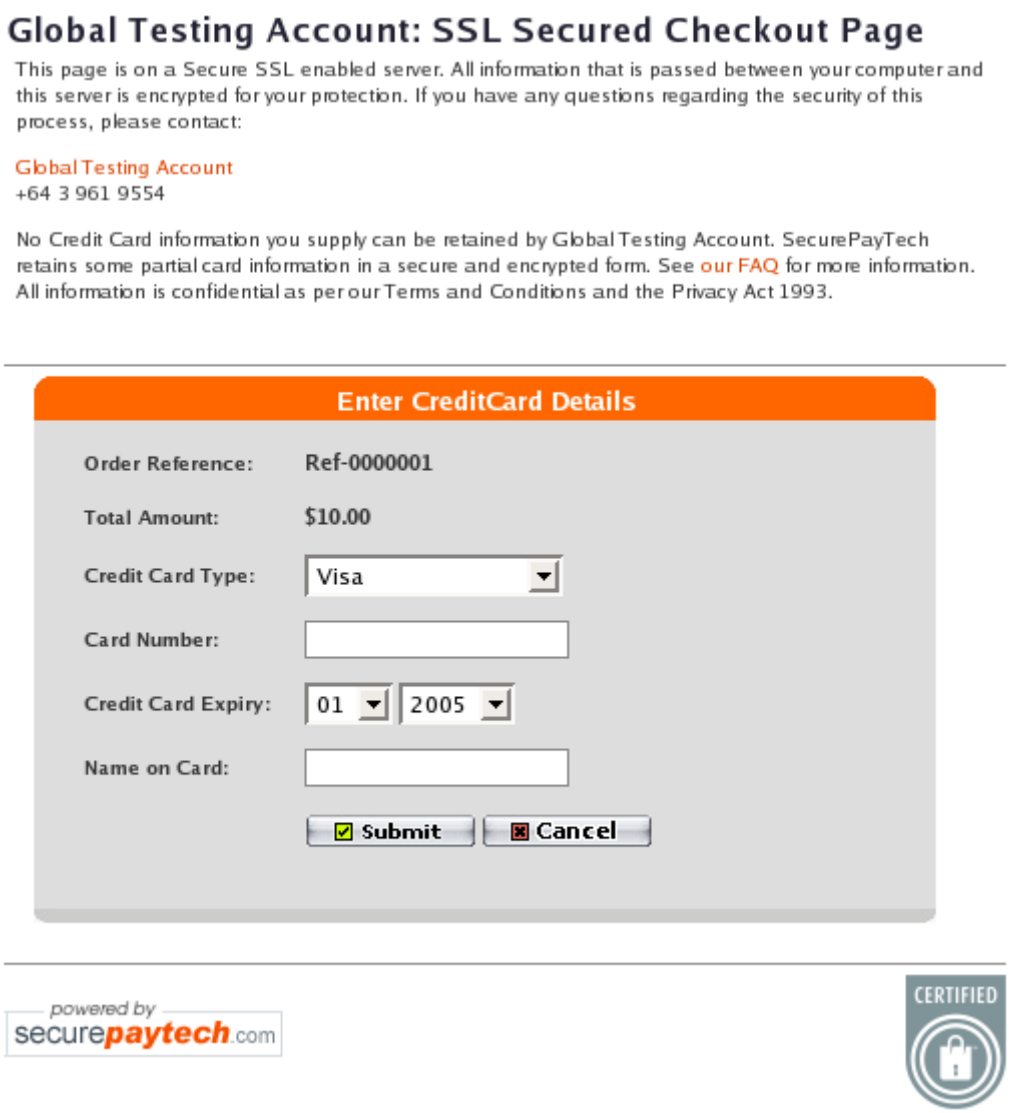

Figure 3: The SecurePayTech.com hosted payment page

Note that the form shows the name of your business (in this example: “Global Testing Account”) and includes your email contact details and phone number (according to your SecurePayTech.com account details).

The customer then enters their card details and clicks on “Submit”:

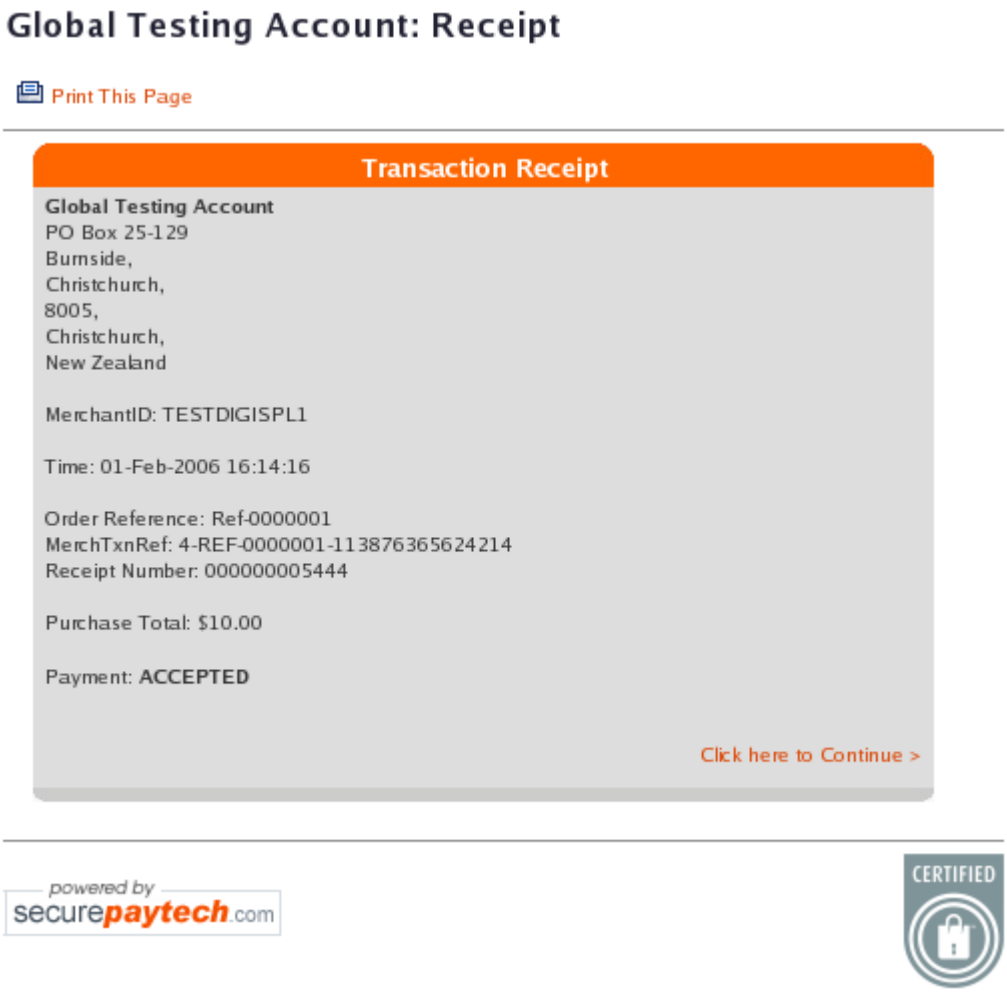

Figure 4: Successful payment receipt page

If the payment was successful, then a page similar to the one above will be shown. It shows your company name and address, as well as details of the transaction itself.
An email notification of the successful purchase will be sent to the contact address set in the SecurePayTech.com account details.
The *“Click here to Continue >”* link will take the customer to the “Return URL” page you have specified in your website's original form submission. ***Note:*** *If you plan on passing variables back to your website via the return URL, be aware that there is nothing forcing the customer to click on the “continue” link*.

Figure 5: Receipt page for a declined transaction

If the payment is declined, then a receipt page like the one in Figure 5 above will be shown. A link is provided to allow the user to go back and re-enter their card details on the previous form (i.e. Figure 3).

Testing Accounts

In order to perform test transactions through the SecurePayTech beta testing system, you will need to be assigned a test “Merchant ID” code, and a “Merchant Transaction Key” Transactions performed by one of these test accounts behave slightly differently from a full, production account, in that the results returned from it are dependant on the monetary amount of the transaction.

Until you are assigned a VPS Merchant ID, you can use the following details to run test transactions through the system. Note that you will be unable to view these transactions through your login on the web interface. Once you have been assigned a VPS Merchant ID, and you start using that, then the web interface will show those transactions.

Temporary Test VPS Merchant ID: **TESTDIGISPL1**

Testing Credit Card Numbers

Card Type	Card Number	Expiry (MM/YY)
Visa	4987-6543-2109-8769	05/13
Mastercard	5123-4567-8901-2346	05/13
American Express	3456-7890-1234-564	05/13
Diners Club	3012-345678-9019	05/13

Testing Dollar Values

Table 2: Transaction Result Codes

The table showing the cent values and test return codes is below:

Cent Value	Response Code	Description
.00	1	Transaction OK
.10	2	Insufficient Funds
.54	3	Card Expired
.57	7	Unsupported Transaction Type
.75	4	Card Declined
.91	6	Communications Error

NOTE: Using any cent value that is *not* .00, will result in a transaction error of some kind (usually a Server Error). In a live, production system, any cent value can be used, it will not effect the result in this way.

Explanation of Fields / Terms

- **VPS MerchantID:** Unique identifier assigned to you which ties transactions to your merchant bank account.
- **OrderReference:** A reference number or string that has meaning in your own application. This does not necessarily need to be unique. Example: “AcmeWidgetOrder_3424”. ***This field should be no longer than around 20-22 characters***. It forms part of the MerchTxnRef, which itself can be no longer than 40 characters. If it needs to, the system will truncate your OrderReference in order to satisfy this.
- **MerchTxnRef:** Unique identifier assigned to each transaction by the SecurePayTech system, and can be used to retrieve information on that transaction later on.
- **ReceiptNo:** Identifier for the receipt data generated by ETSL. Also known as the RRN.

Custom Hosted Payment Page

The custom hosted payment page enables merchants to customise SecurePayTech's hosted payment page with styling that is consistent with the merchant's website. This is to provide a consistent look and feel to customers during the purchasing process and to avoid any security concerns a customer may have when entering their credit card details.

Merchants who wish to utilise this facility must inform SecurePayTech of their wishes so we can enable this feature.

Custom Pages

Merchants will have to provide us a compressed archive file (in ZIP or TAR formats) containing:

- /images/
- /javascripts/
- /stylesheets/
- index.html
- pay.html

Functionality

The custom hosted payment page functionality accepts the same parameters as the generic hosted payment page with the additional 'useCustom' parameter. If this parameter is not specified in the request, it will default to the generic hosted payment page.

```
<form action="https://merchant.securepaytech.com/paymentpage/index.php"
method="post" id="mypaymentform">
 <input type="hidden" name="amount" value="10.00" />
 <input type="hidden" name="merchantID" value="TESTDIGISPL1" />
 <input type="hidden" name="returnURL"
       value="http://www.youronlinestore.co.nz/cart/thankyou.html" />
 <input type="hidden" name="cancelURL"
       value="http://www.youronlinestore.co.nz/cart/pre-payment.html" />
 <input type="hidden" name="orderReference" value="Ref-0000001"/>
 <input type="hidden" name="enableCsc"/>
 <input type="hidden" name="useCustom"/>
 <input id="sub" type="submit"  value="Pay by Credit Card" />

</form>
```

Index page

The index.html file will be the template that will be rendered when a customer is redirected to SecurePayTech's hosted payment page. Any stylesheets, images, javascript source files included in the ZIP or TAR file may be sourced from this page. The index file must display the purchase details, including amount, order reference as well as the credit card form to capture details required to process the payment.

Sample index.html page:

```
<!DOCTYPE html
 PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
               "http://www.w3.org/TR/xhtml1/DTD/xhtml-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
  <head>
    <link rel="stylesheet" type="text/css" href="${cssroot}/default.css" />
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
    <title>Merchant X Payment Page</title>
  </head>
  <body onLoad="javascript:this.document.payform.reset();">
  <p> This is an example custom hosted payment page for SecurePayTech and is hosted on an SSL
      enabled server. </p>

  <form method="post" action="pay.php" name="payform">
    <input type="hidden" name="merchantID" value="{$merchantID}" />
    <input type="hidden" name="returnURL" value="{$returnURL}" />
    <input type="hidden" name="orderReference" value="{$orderReference}" />
    <input type="hidden" name="amount" value="{$amount}" />
    <input type="hidden" name="enableCsc" value="{$enableCsc}" />
    <input type="hidden" name="useCustom" value="{$useCustom}" />
    <p>Order Reference: {$orderReference}</p>
    <p>Total Amount: ${$amount}</p>
    <p>Credit Card Type:
      <select name="ccType">
        <option value="1">Visa</option>
        <option value="2">Mastercard</option>
        <option value="3">American Express</option>
        <option value="4">Diners Club Card</option>
      </select>
    </p>
    <p>Card Number:
      <input type="text" name="ccNumber" maxlength="32" size="20" />
    </p>
```

```
<p>Credit Card Expiry:
      <select name="expiryMonth">
        <option value="01">01</option>
        <option value="02">02</option>
        <option value="03">03</option>
        <option value="04">04</option>
        <option value="05">05</option>
        <option value="06">06</option>
        <option value="07">07</option>
        <option value="08">08</option>
        <option value="09">09</option>
        <option value="10">10</option>
        <option value="11">11</option>
        <option value="12">12</option>
      </select>

      <select name="expiryYear">
        <option value="09">2009</option>
        <option value="10">2010</option>
        <option value="11">2011</option>
        <option value="12">2012</option>
        <option value="13">2013</option>
        <option value="14">2014</option>
        <option value="05">2015</option>
      </select>
    </p>

    {if isset($enableCsc) }
      <p>Card Security Code:
         <input type="text" name="ccCsc" maxlength="4" size="4" value="{$ccCsc}" />
      </p>
    {/if}

    <p>Name on Card:
      <input type="text" name="ccName" maxlength="20" size="20" value="{$ccName}" />
    </p>
    <input id="submitbutton" type="submit" name="submit" onClick="this.disable = true" />
  </form>

 </body>
</html>
```

Note: Example above is shown without any form field validation.

Payment Receipt Page

The payment receipt page (pay.html) which will display the outcome of the payment can also be customised. Links that allow customers to retry their payment or redirect them back to the merchants website should be placed here (see example below).

Sample pay.html page

```
<!DOCTYPE html
     PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <link rel="stylesheet" type="text/css" href="${cssroot}/default.css" />
  <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
  <title>Payment Result</title>
</head>

<body>
  <h1>{$businessName}: Receipt</h1>
  <p>{$receiptOutput}</p>
  {if $outcome == "ACCEPTED"}
    <p>
    <a href="{$returnURL}">To complete your order you <b>must</b> click here to continue &gt;</a>
    </p>
  {else}
    <p>
      Sorry, <a href="javascript:history.go(-1)">Go back and try again</a>
    </p>
  {/if}
</body>

</html>
```