

Stellar Web Solutions eVendor V1.1 (for PayPal)

Issued: 2007 07 31

Overview

Congratulations on the purchase of the Stellar Web Solutions eVendor system. The Stellar Web Solutions eVendor allows you to sell and deliver electronic goods such as eBooks, music, or software from your own website. This version of the eVendor supports single item transactions, for selling multiple items in the same transaction the eVendor Pro version is required. By using this software, you agree to the conditions of the attached licensing agreement. This software is licensed - not sold, please read the license agreement carefully. This software is protected by international copyright, it cannot be redistributed, resold, or transferred in whole or in part in any way.

Requirements

- This software is designed to work on both Linux/UNIX or Microsoft Windows.
- Web Server with a PHP 4 or higher.
- Database – MySQL 4 or higher is recommended.
See www.mysql.com for more information about MySQL.
- For Encrypted Website Payments (EWP) OpenSSL 0.93 or higher is required.

Package Manifest

The software distribution package contains the following files:

- readme.pdf – this file
- license.pdf – software license.
- install.php – Database installation script.
- ipn.php – PayPal IPN Script.
- stellar-settings.php – Configuration options.
- stellar-ipnmod.php – IPN code library.
- stellar-vendor.php – Sales and electronic delivery library.
- return.php – Sample return/download page.
- checkout.php – Sample checkout page.
- index.php – Sample integration page – sample purchase links.

Support

To obtain installation support, we recommend emailing us at stellar@stellarwebsolutions.com. Otherwise, visit www.stellarwebsolutions.com/en/contact.php for more ways to get support.

Installation Services

Contact Stellar Web Solutions to inquire about our low cost professional installation services available.

Features

Integrates PayPal processed transactions to securely deliver an electronic product that can be downloaded by your customers. PayPal IPN data is authenticated, verified, and stored in a MySQL database.

Secure Anti-Tampering System

- Transaction data is authenticated by retransmitting the received data back to PayPal to verify it's authenticity.
- Verification of the transaction being sent to your PayPal account and not spoofed.
- Transactions are only processed one time, duplicate retransmissions are ignored.
- Pending transactions will not allow a download until the funds are received.
- Access to disputed or refunded orders is instantly removed.
- Download user's IP address is recorded, optional email notification of each download.
- Purchaser's email address is used to access the download link to deter link sharing.

Secure Anti-eShoplifting System

- Optional encrypted websites payments (EWP) to encrypt pricing and options prior to transmission to PayPal – see http://www.stellarwebsolutions.com/en/articles/paypal_button_encryption_php.php for instructions to create the necessary certificates to support encryption and for instructions to add the certificates to your PayPal account.

PHP Primer

If you are not completely familiar with the PHP programming language, the website php.org is a valuable reference. While our software is designed to not require extensive PHP experience, some basic knowledge is necessary.

Syntax

PHP code is embedded into HTML, written within “<?PHP” and “?” tags, PHP code is executed by the webserver prior to sending the result to a user. This allows passwords and confidential information to be used – such as database connection settings – as this “code” content cannot be seen by users visiting your website. Each line of PHP code

must end with a semi-colon “;” - the exception to this is that strings can run over multiple lines. Normal strings are enclosed within single quotes (apostrophes) 'string' and strings which may contain variables must be enclosed within double quotes “string with a \$variable”.

Comments

Text that follows slashes “//” on a line are not executed. We typically comment areas of the code where an advanced user might wish to change to the look and feel of the output of the scripts.

Variables

Variables are defined with a dollar sign, such as \$variable. A variable followed by an equals sign assigns a value to the variable. When configuring the stellar-settings.php file you will assign variables to values. Values within single quotes (apostrophes) 'value' or double quotes “value” are strings. Quotes are not required when setting a value to the value returned from a function.

Sample variable assignment:

```
$variable = 'text value';
```

Arrays

Similar to normal variables, arrays add an extra level of complexity and allow multiple items to be stored within the same variable. Most of the Stellar Web Solutions code uses associative arrays which use two entries – a name and value – for each item in the array. This allows the value that corresponds to a name to be found easily. Items are assigned within the “array()” function with the “=>” operator and separated with commas “,”.

Sample array definition:

```
$array = array('name1' => 'value1',  
              'name2' => 'value2',  
              );
```

You can add items to an already defined arrays as follows:

```
$array['name3'] = 'value3';
```

Displaying Text

To display text output on the webpage – use the “echo” function. To include a new line, use “\n”.

Example:

```
echo “this is a string\n”;
```

Block of text spanning multiple lines:

```
echo “<P>block of text  
spanning multiple  
lines</P>”;
```

System Overview

The eVendor system begins with the checkout page which formats a purchase form for transmission to PayPal. You integrate a button anywhere on your website(s) to point to the checkout.php page (Step 4), this page automatically builds either an encrypted or unencrypted PayPal button and automatically submits the form to PayPal. The customer proceeds with the purchase on PayPal's website and follows a return link to your website. An IPN post from PayPal to ipn.php uploads the transaction data to your website and the data is stored in your MySQL database. The return.php page displays a download link for the customer after payment was successful. The IPN processing script also emails an alternate download link in case the customer did not return to your website or paid with a non-instant payment that requires them to check back later. A refund or disputed transaction automatically cancels the associated download links.

Installation Procedure

To install the Stellar Web Solutions eVendor and IPN scripts there is a four step process:

1. Copy the PHP files to your website.
2. Edit the stellar-settings.php PHP file to customize your site settings and password. You will need to know the database settings from your ISP – database hostname/IP, username, password, and the database name – called a schema.
3. In your web browser, visit install.php to build the database tables.
4. Set up your website to call the checkout PHP functions.
5. Customize the look of your checkout/download pages.

Install Step 1: Installing the PHP Files

Prior to this step, you should have a webserver which can execute PHP code, you should have at least PHP Version 4 or higher. Most commercial hosting services provide PHP already installed.

- A. Place the provided .php files on your webserver in a web accessible directory. You can use FTP or a method specified by your internet service provider to upload files to your webserver.

Install Step 2: Configuring stellar-settings.php text file.

The stellar-settings.php file is a PHP “include” file with the common variables used by the other PHP scripts. This file can be edited with any text editor, if you cannot edit files right on your webserver, FTP the file to a PC where you can use Notepad/Wordpad (or equivalent). Please set values to the following system variables:

System Variables Configuration

The format of the each variable is “\$variable_name = 'value'; //comment”:

- \$db_host = 'localhost'; // database host, if local enter localhost
- \$db_schema = 'schema'; // database schema i.e. Name of database

- \$db_user = 'user'; // database username to add, update, remove data, add tables.
- \$db_password = 'password'; // database password for the above user
- \$business = 'email@website.com'; // your PayPal primary email address
- \$currency_code = 'USD'; //PayPal currency code for buttons - United States=USD, Canada=CAD, Great Britain=GBP, Euros=EUR etc.

The \$paypal_cart array contains fields passed directly to PayPal – the notify_url and return page addresses are required:

```
//settings that will be sent directly to PayPal
$paypal_cart = array(
    'cbt' => 'Get the Instant Download Now',
    'notify_url' => 'http://www.yoursite.com/shop/ipn.php',
    'return' => 'http://www.yoursite.com/shop/return.php',
    'rm' => '2',
);
```

Advanced Settings

- \$db_table_sold_trans = 'stellar_shop_trans'; //database table for transactions
- \$db_table_sold_item = 'stellar_shop_item'; //database table for items
- \$db_table_sold_download = 'stellar_shop_download'; //database table for download tracking
- \$receiver_id = 'XXXXXXXXXXXX'; //optional – validate transactions based on receiver ID instead of email address (not recommended)

Encryption Settings (only required to use PayPal button encryption)

- \$use_encryption = 0;
- \$site_key = '/usr/home/stellar/paypal/prvkey.pem';
- \$site_cert = '/usr/home/stellar/paypal/pubcert.pem';
- \$site_cert_id = '1234567890HSU';
- \$paypal_cert = '/usr/home/stellar/paypal/paypal_cert.pem';
- \$sandbox_cert = '/usr/home/stellar/paypal/paypal_sandbox.pem';
- \$site_sandbox_key = \$site_key;
- \$site_sandbox_cert = \$site_cert;
- \$site_sandbox_cert_id = '12345678906EA';
- \$OPENSSL = "/usr/bin/openssl";

Install Step 3: Building the Database Table.

To install the tables into your database you will need to open your web browser to your website and the path of the install.php file. The database will be automatically be installed if your settings in stellar-settings.php were correct.

Example: <http://www.stellarwebsolutions.com/shop/install.php>

If you receive an error message, you are likely missing the correct, hostname, username, password, or database schema (database name).

Install Step 4: Setup and use eVendor Digital Goods delivery system.

PHP Item Catalog Setup

The PHP catalog defines both pricing and product information for checkout to PayPal, as well as the download filename for electronic goods delivery once the customer has paid. Items in the catalog.php are stored in a PHP array format, the provided catalog.php file contains sample item entries in an array called \$items which is an array of associative arrays.

```
array(
  'item_name' => 'Item Name',
  'item_number' => 'Item Number',
  'amount' => '123.45',
  'on0' => 'First Option Name',
  'os0' => 'First Option Selection',
  'on1' => 'Second Option Name',
  'os1' => 'Second Option Selection',
  'file' => 'package.zip',//filename of product – can be any format
),
array(
  'item_name' => 'Item Name 2',
  'item_number' => 'Item Number 2',
  'amount' => '123.45',
  'on0' => 'First Option Name',
  'os0' => 'First Option Selection',
  'on1' => 'Second Option Name',
  'os1' => 'Second Option Selection',
  'file' => 'package2.mp3',//filename of product – can be any format
),
```

You can optionally use dynamic PHP code to create the list of items from calculated values or from a database.

All of the above variables are standard PayPal variables.

Email Confirmation

If the setting \$global_do_email=1; in stellar-settings.php, a confirmation email will be sent to the purchaser on successful receipt of a IPN transmission from PayPal. The contents of the email should be set in the sendConfirmation function in stellar-settings.php.

Example:

```
in stellar-settings.php:
    $global_do_email=1;
```

```

function sendConfirmation($to, $from, $txn_id, $custom, $invoice)
{
    #email confirmation
    $subject = "Purchase Download Instructions - StellarWebSolutions.com";
    $content = "Thank you for you purchase on StellarWebSolutions.com. \n"
        ."In case you were not automatically redirected to the\n"
        ."instant download page after completing the purchase,\n"
        ."you may access your downloads from the following page:\n\n"
        ."http://www.stellarwebsolutions.com/shop/return.php?payer_"
        ."email=$to&txn_id=$txn_id\n"
        ."Purchaser Email: $to\nTransaction ID: $txn_id\n\n"
        .listItems($txn_id)
        ."For technical support questions, please email “"
        ."stellar@stellarwebsolutions.com.\n\n"
        ."www.stellarwebsolutions.com";
    emailMessage($to,$from,$subject,$content);
};

```

Note: The listItems(\$txn_id) function will optionally list all the items within the order.

Purchase Links and Buttons for the Checkout Page

Items can checked-out to PayPal via the Stellar eVendor with an HTML form “POST” or added to the URL (GET request). Buttons or get requests should point to your checkout page, ie the checkout.php. You do not need to use the included checkout page for integration into your own shopping cart system – just remember to use the eVendor's return URL and notify_url variables to ensure the transaction data is posted to the eVendor's IPN script and that the product item_number and file are defined in the catalog.php file.

Fields accepted by the shopping cart page are:

- item_number – the item number
- quantity – number of items – if left blank defaults to 1
- on0 and on1 – option names
- os0 and os1 – option choice

Basic add to cart button example:

```

<form action="checkout.php" method="POST">
<input type=hidden name="item_number" value="widgetproduct-02">
<input type=submit name="Submit" value="Buy Our Widget eBook $29.99">
</form>

```

Example GET requests:

Buy an item:

http://yoursite.com/shop/checkout.php?item_number=widgetproduct-07

Congratulations, you have now installed the Stellar Web Solutions eVendor system.