

Introduction

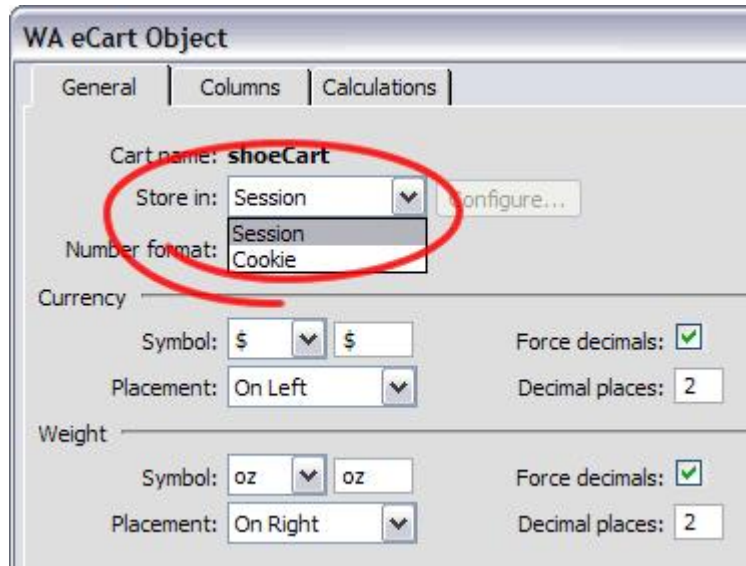
WA eCart users are able to keep shoppers on their eCommerce sites without passing them to a payment gateway site: we call this process local checkout. This requires using an Secure Socket Layer (SSL) server to allow customers to enter their credit card information securely. In some cases, shopping cart data is cleared when the SSL server is contacted. This TechNote discusses why that happens and describes how to get your data to the secure area of your site so that your transactions can be processed properly.

Sensitive information and the reason to use SSL

SSL is to add security to your customer to prevent someone from listening in on the information provided in the form. SSL is necessary in order to give your consumer confidence when buying from your web site any time you have them enter their credit card number, social security number, or any other highly sensitive information. No web savvy customer will enter a credit card number on a web site without SSL encryption.

Understanding the problem

WA eCart can be set up to store the shopping cart as a cookie or as a session variable. These are the most common shopping cart storage mechanisms, but they both depend on the web browser's underlying cookie mechanism to track the user so that shopping carts contents can be maintained for that user. It's a little-understood fact that even session variables use cookies.



Cookies are stored on your local computer hard drive with the domain name and cookie name-value pairs. The domain name is used as a qualifier to give the browser access to those variables for security reasons. In the case of using Session variables, only an ID is stored as a cookie and all of the name-value pairs are stored on the server with the associated ID as a look-up value. The session variable is generally considered more secure and has a larger data storage capacity, but has a more limited time range for storage.

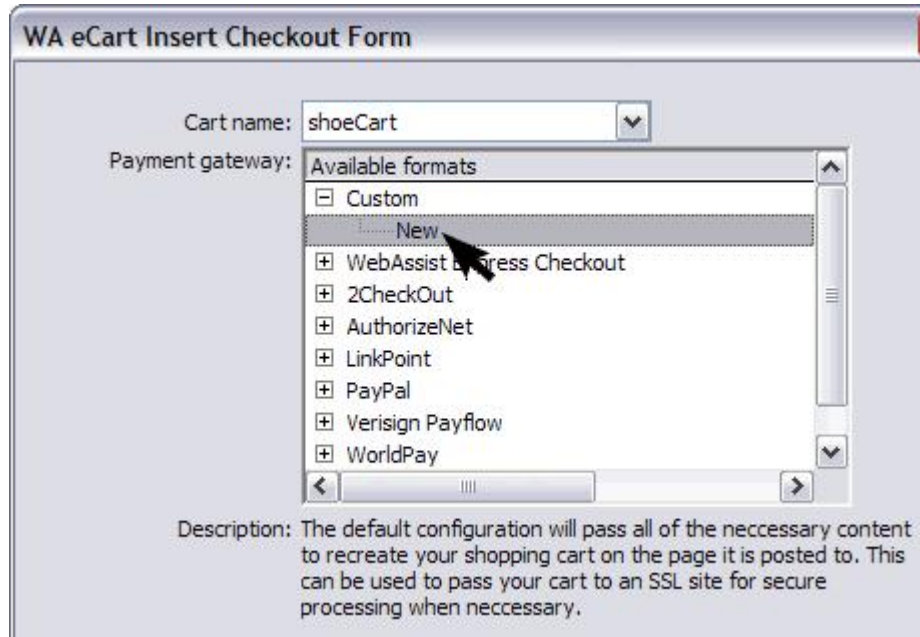
In order to process credit card transactions securely, the merchant must obtain an SSL certificate which has server grade cryptology capabilities for their site. The Web server administrator must install the certificate on the Web server and create a directory on the server for storing pages that will receive encrypted data. When your SSL directory is created it may or may not use the same domain name. For example you may have your eCart application on your store at <http://www.yourStore.com> once your SSL server is installed it may be locate at the same domain like <https://www.yourStore.com>, or it could also be set up as a new domain like: <https://www.yourSecureStore.com>. If your standard site and your secure site are the same, this TechNote will not apply to you. The data is stored based on the domain so the same cookies and session will be shared. However, if your standard site and SSL-protected site have different domain names, you'll need to follow this TechNote to learn how to transfer your shopping cart from one domain to another.

The solution

The data transfer of a shopping cart from one site to another is achieved by creating a special form with hidden form elements containing all of the shopping cart data which is then posted to the new domain. The new domain reads in the form and repopulates the shopping cart based on its contents. This is very similar to how payment gateways such as PayPal and WA Express Checkout allow you to pass data to their shopping carts. That is why they use the same "Insert Checkout Form" object on the WA eCart insert bar to insert a custom checkout form:



To start, expand the Custom option and click on New from the list:



Next, click the Configure button at the bottom of the WA eCart Insert Checkout Form dialog:



In the Form action field of the Configure Checkout Form dialog, enter the URL to the page on your secure server you'd like to POST your shopping cart details and any other information.

Configure Checkout Form

Label:


Name:

Default:

Display as:

Label	Name	Default	Display
Cart Column 'ID'	ID_{x}	<?php echo \$shoeCart->Dis...	hidden
Cart Column 'Name'	Name_{x}	<?php echo \$shoeCart->Dis...	hidden
Cart Column 'Description'	Description_{x}	<?php echo \$shoeCart->Dis...	hidden
Cart Column 'Weight'	Weight_{x}	<?php echo \$shoeCart->Dis...	hidden
Cart Column 'Quantity'	Quantity_{x}	<?php echo \$shoeCart->Dis...	hidden
Cart Column 'Price'	Price_{x}	<?php echo \$shoeCart->Dis...	hidden

Form action:

WebAssist 

Click OK. An alert will appear asking you to save your settings, click OK to save your settings.

Macromedia Dreamweaver

 This configuration has been changed.
It must be saved before it can be used.

Name your saved settings and click OK.

Save Payment Gateway Configuration

Save as:

On the WA eCart Insert Checkout Form dialog, click Done.

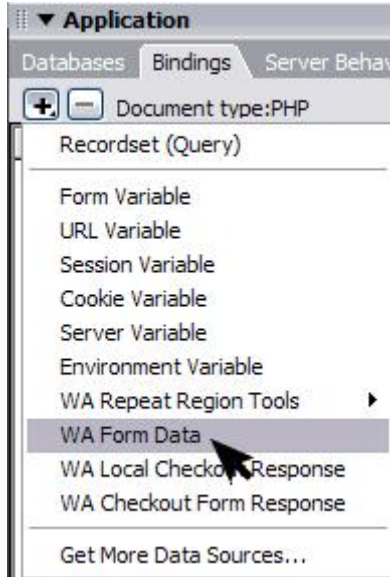
A form with hidden form elements will be placed on the page. You can add to that form more hidden form elements to account for other values you might need passed to the new domain other than the contents of the cart. For instance user information from your database you want to use to pre-fill the form or a UserID, or perhaps other session variables you are using in calculations.

This page is now complete and can be saved and closed. You are ready to start working on your secured page.

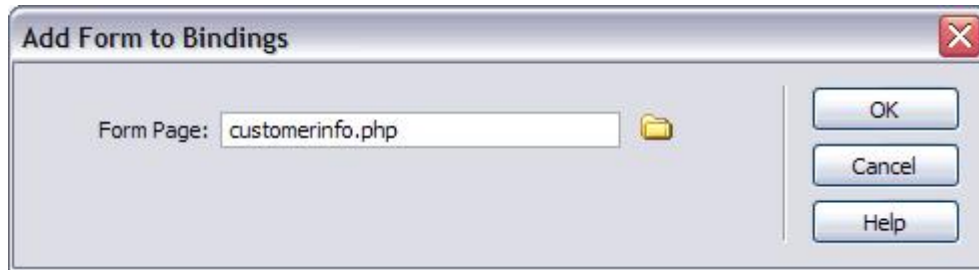
Secure Server Page

The page on the secure server receiving your posted data must be configured to accept the incoming data. To do this, you can use the WA eCart Get Contents From Form server behavior, as described later in this TechNote. First, however, it is helpful to bring in the bindings from the form on the previous page for reference.

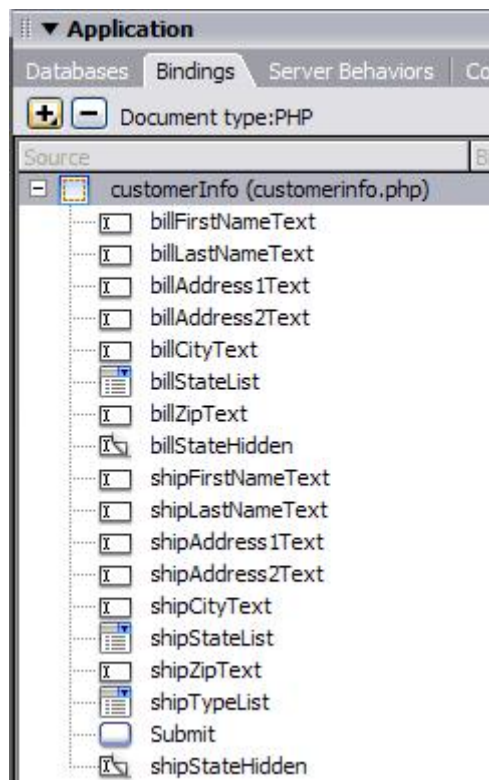
From the Bindings panel, click on the Add (+) button and choose WA Form Data from the list.



In the Add Form to Bindings dialog, select the Browse icon and choose the Customer Info page from your non-secure site.



The Bindings panel will now have a new item you can expand to obtain bindings for the form data posted from the Customer Info page:



Now you are ready for the final step. On the secure server page, open the WA eCart Object panel and add a cart object to the page, if one does not exist; Remember that the WA_eCart folder and its contents must be included with this file when you upload it to any server.

In the Server Behaviors panel, choose Add (+) and select WA eCart > Add To Cart > Get Contents From Form.

Get Contents From Form

Trigger: any form post

Cart name: shoeCart

Payment gateway: Custom Configure...

Description: Creates server side code to capture cart details from another website.

OK
Cancel
Help

Since the only way this page can be accessed is from the form POSTed on the Custom Checkout page, use "any form post" as the trigger. Then, select the cart name and choose Custom from the Payment gateway list. Click Configure to view the values that will be loaded into the shopping cart object.

Configure Transaction Details

Cart Column	Value
ID	<?PHP echo \$_POST["ID_".(\$displayIndex)]?>
Name	<?PHP echo \$_POST["Name_".(\$displayIndex)]?>
Description	<?PHP echo \$_POST["Description_".(\$displayInd...]
Weight	<?PHP echo \$_POST["Weight_".(\$displayIndex)]?>
Quantity	<?PHP echo \$_POST["Quantity_".(\$displayIndex...]
Price	<?PHP echo \$_POST["Price_".(\$displayIndex)]?>

Value: <?PHP echo \$_POST["ID_".(\$displayIndex)]?> Update

OK
Cancel
Help

You should not need to make any changes, so click OK and save the configuration. Enter the configuration name, click OK to save, then click OK again.

As before, bring up the Bindings panel and click on the Add (+) button. Choose WA Form Data and click the Browse icon to select the Custom Checkout page from your site. This will add bindings for all of the additional

items and values you created by adding hidden form elements to the custom checkout form.

With the bindings from your Custom Checkout form exposed on the secure server page, you can now use these values to pre-fill form elements, pass data to a payment gateway, store data in your database or set session variables just as you can do with in the non-secure WA eCart pages.