Alternative CVV2 flow for the WebSphere Commerce V7 Madisons starter store

Skill Level: Introductory

Dilip H. Ramesh (dilip.ramesh@in.ibm.com)
Software Developer
IBM

Pavan Kini (pavan.kini@in.ibm.com)
Software Developer
IBM

17 Mar 2010

This article explains how to make changes to the Madisons store to capture CVV2 input just before placing an order.

Introduction

CVV2 stands for Card Verification Value. It is a 3 or 4 digit number used for secure online credit card transactions. In the Madisons out-of-the-box store, CVV2 input is asked for on the Shipping & Billing Method page. The masked out CVV2 value will be displayed on the Order Summary page before submitting the order. This article explains an alternative flow where CVV2 is asked for on the Order Summary page.

Step 1

Remove the code snippet related to CVV2 from the following files. This removes the CVV2 field in the Shipping & Billing Method page.

Refer to `StandardVisa.jsp` in the code_snippets.zip file that is provided in the Download section of the article.

Path

```
/<storedir>/Snippets/EDP/PaymentMethods/StandardVisa.jsp
/<storedir>/Snippets/EDP/PaymentMethods/StandardMasterCard.jsp
/<storedir>/Snippets/EDP/PaymentMethods/StandardAmex.jsp
/<storedir>/Snippets/EDP/PaymentMethods/StandardDiscover.jsp
```

`<storedir>` is the name of the directory where store-specific assets are located. For example, `Madisons, MadisonsStorefrontAssetStore`.

Step 2

Add **CVV2** as an input field in the Order Summary page.

Refer to `CheckoutPaymentAndBillingAddressSummary.jspf` in the code_snippets.zip file that is provided with this article.

Path

```
/<storedir>/ShoppingArea/CheckoutSection/CheckoutPaymentAndBillingAddressSummary.jspf
```

Step 3

Add a new WebSphere Commerce Ajax Service declaration in ServicesDeclaration.js as shown in Listing 1.

Path

```
/<storedir>/javascript/ServicesDeclaration.js
```

Listing 1. Service declaration

```
/**
* Declares an AJAX service that updates payment instruction in the
current order.
* @constructor
*/
wc.service.declare({
        id: "AjaxUpdatePaymentInstructionInSummary",
        actionId: "AjaxUpdatePaymentInstructionInSummary",
        url: "AjaxOrderChangeServicePIUpdate",
        formId: ""

        /**
        * On success, calls updateCVV2AndCheckoutOrder() function.
        * @param (object) serviceResponse The service response object,
which is the JSON
    * object returned by the service invocation.
         */

        ,successHandler: function(serviceResponse) {
                CheckoutHelperJS.updateCVV2AndCheckoutOrder();
        }

        /**
         * Displays the error message returned with the service
response.
         * @param (object) serviceResponse The service response object,
which is the JSON
     * object returned by the service invocation.
         */
        ,failureHandler: function(serviceResponse) {
                //Failure handler code goes here
        }
}),
```

This service is used to incorporate CVV2 to a Payment method in the Order Summary page. Multiple payment methods are also supported.

`com.ibm.commerce.edp.commands.PIEditCmd` is the command invoked for updating a Payment Instruction (PI). The default command implementation of `PIEditCmd` requires you to pass in all protocol data (except `piAmount`) when editing a payment instruction.

However, to update only CVV2, you can change the implementation class from the default of `com.ibm.commerce.edp.commands.PIEditCmdImpl` to `com.ibm.commerce.edp.commands.PIEditWithMergingProtocolDataCmdImpl` in the database table CMDREG. This is another implementation class provided by WebSphere Commerce, which does not require you to pass in all protocol data. Only the parameters passed to the `PIEditCmd` command will be treated as changed. Parameters that are not passed remain unchanged in the payment instruction.

Step 4

Add a new function in CheckoutHelper.js to invoke the above service as shown in Listing 2.

Path

```
/<storedir>/javascript/CheckoutArea/CheckoutHelper.js
```

Listing 2. Service invocation

```
/**
* This function checks for PIs that need to be updated with CVV2 and
invokes the
* "AjaxUpdatePaymentInstructionInSummary" service,
*/
updateCVV2AndCheckoutOrder:function(){
  if(this.paymentsToUpdate <= this.numberOfPaymentMethods){
    var i = this.paymentsToUpdate;
    this.paymentsToUpdate = this.paymentsToUpdate + 1;
    if(document.getElementById("cc_cvc_" + i)!= null){
    var params = [];
    params["payMethodId"] = document.getElementById("paymentMethodId_" +
i).value;
    params["piId"] = document.getElementById("paymentInstance_" +
i).value;
    params["authToken"] = document.getElementById("authToken").value;
    params["cc_cvc"] = document.getElementById("cc_cvc_" + i).value;
    wc.service.invoke('AjaxUpdatePaymentInstructionInSummary',params);
  }
  else{
    this.updateCVV2AndCheckoutOrder();
  }
}
else{
this.checkoutOrder(this.orderId,this.userType,this.addressListForMailNotification,
 this.isQuote);
}
},
```

Step 5

Modify the Single Shipment Summary page and the Multiple Shipment Summary page to use the new function `updateCVV2AndCheckoutOrder()`.

Refer to `SingleShipmentSummary.jsp` and `MultipleShipmentSummary.jsp` in the code_snippets.zip file that is provided with the article.

Path

```
/<storedir>/ShoppingArea/CheckoutSection/SingleShipment/SingleShipmentSummary.jsp
/<storedir>/ShoppingArea/CheckoutSection/MultipleShipment/MultipleShipmentSummary.jsp
```

Following are the prerequisites for invoking `updateCVV2AndCheckoutOrder()function`:

1. Set the parameters `orderId`, `userType`, `addressListForMailNotification`, `isQuote` and `numberOfPaymentMethods` for the `ChekoutHelperJS` class. These parameters are required for the `checkoutOrder()` function, which is invoked after updating CVV2.

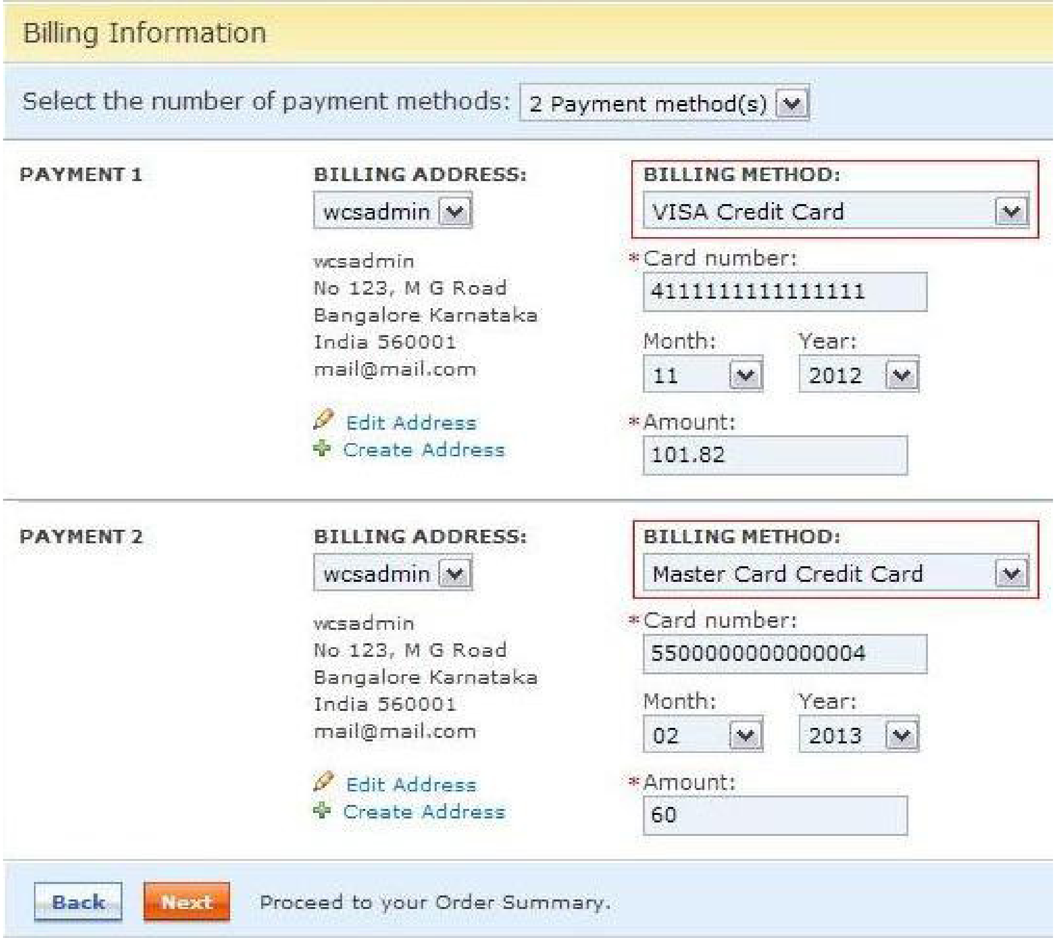
2. Validate that the mandatory CVV2 value is numeric.

Testing your changes

1. In the Madisons store, add items to the shopping cart and click **Checkout**. You will be redirected to the Shipping & Billing Method page. As shown in Figure 1, select two Payment Methods: **VISA® Credit Card**

and **Master® Card Credit Card**. Click **Next** to proceed to the Order Summary page.

Figure 1. Billing Information section in the Shipping & Billing Method page



Billing Information

Select the number of payment methods: 2 Payment method(s)

PAYMENT 1

BILLING ADDRESS: wcsadmin

wcsadmin
No 123, M G Road
Bangalore Karnataka
India 560001
mail@mail.com

Edit Address
Create Address

BILLING METHOD: VISA Credit Card

*Card number: 4111111111111111

Month: 11 Year: 2012

*Amount: 101.82

PAYMENT 2

BILLING ADDRESS: wcsadmin

wcsadmin
No 123, M G Road
Bangalore Karnataka
India 560001
mail@mail.com

Edit Address
Create Address

BILLING METHOD: Master Card Credit Card

*Card number: 5500000000000004

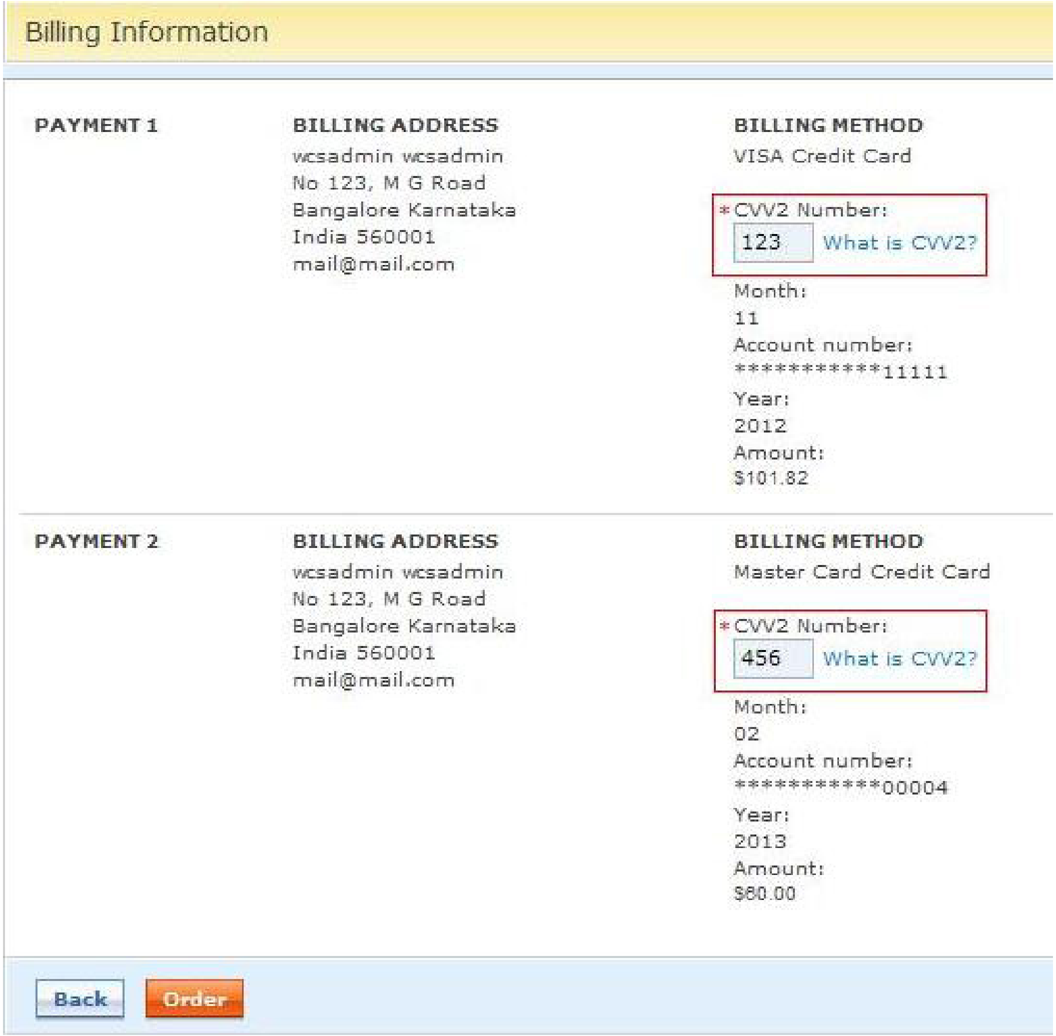
Month: 02 Year: 2013

*Amount: 60

Proceed to your Order Summary.

2. Enter the CVV2 value for each of the Payment methods and click **Order** to proceed to the Order Confirmation page as shown in Figure 2.

   Figure 2. Billing Information section in the Order summary page



Billing Information

PAYMENT 1

BILLING ADDRESS
wcsadmin wcsadmin
No 123, M G Road
Bangalore Karnataka
India 560001
mail@mail.com

BILLING METHOD
VISA Credit Card
* CVV2 Number:
123
Month:
11
Account number:
***********11111
Year:
2012
Amount:
$101.82

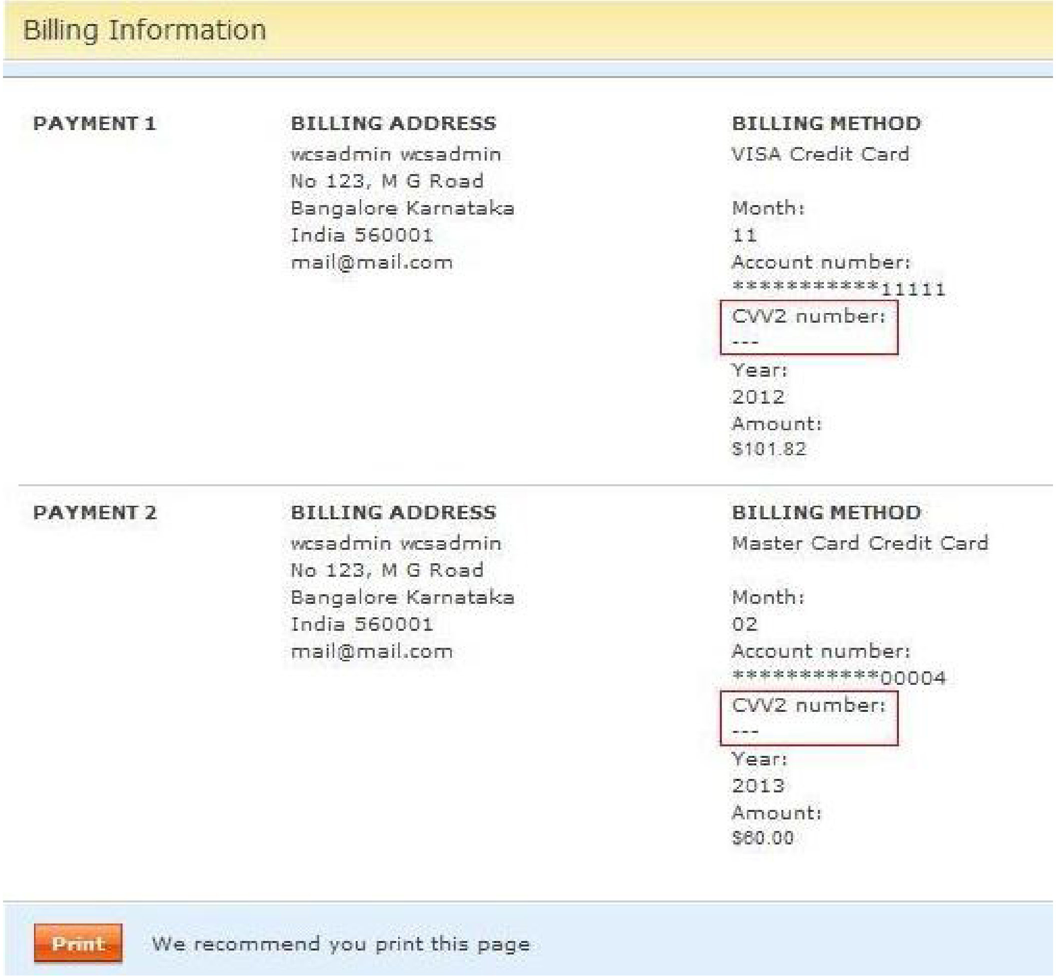
PAYMENT 2

BILLING ADDRESS
wcsadmin wcsadmin
No 123, M G Road
Bangalore Karnataka
India 560001
mail@mail.com

BILLING METHOD
Master Card Credit Card
* CVV2 Number:
456
Month:
02
Account number:
***********00004
Year:
2013
Amount:
$80.00

3. Figure 3 shows that the CVV2 value is now part of the Payment method. The actual value is not displayed for security reasons.

Figure 3. Billing Information section in the Order confirmation page



Billing Information

PAYMENT 1

BILLING ADDRESS
wcsadmin wcsadmin
No 123, M G Road
Bangalore Karnataka
India 560001
mail@mail.com

BILLING METHOD
VISA Credit Card

Month:
11
Account number:
***********11111
CVV2 number:

Year:
2012
Amount:
$101.82

PAYMENT 2

BILLING ADDRESS
wcsadmin wcsadmin
No 123, M G Road
Bangalore Karnataka
India 560001
mail@mail.com

BILLING METHOD
Master Card Credit Card

Month:
02
Account number:
***********00004
CVV2 number:

Year:
2013
Amount:
$60.00

We recommend you print this page

Conclusion

In this article, you learned how to make some simple changes to the Madisons store to capture CVV2 input just before placing an order.

Downloads

Description	Name	Size	Download method
Code sample	code_snippets.zip	19KB	HTTP

Information about download methods

Resources

- WebSphere Commerce V7.0 Information Center
- WebSphere Commerce discussion forum

About the authors

Dilip H. Ramesh

Dilip H. Ramesh is a Software Developer for the WebSphere Commerce store models in IBM India. He has one and a half years of experience in the e-commerce field. His areas of experience include J2EE, Ajax, and Dojo.

Pavan Kini

Pavan Kini is a Software Developer for the WebSphere Commerce store models in IBM India. He has one and a half years of experience in the e-commerce field. His areas of experience include J2EE, Ajax, and Dojo.