**JSP 3 Party (Server Hosted) VPC SHA256 Example Code**

## What files are included in the package?

* JSP\_VPC\_3\_Party.html
  + Displays the input fields required to complete the transaction
* JSP\_VPC\_3\_Party\_DO.jsp
  + Sorts POST data, constructs SHA256 HMAC and sends Digital Order to the Payment Server
* JSP\_VPC\_3Party\_Order\_DR.php
  + Displays Digital Receipt data

## How do I change from MD5 to SHA 256?

Get your developer to look at your current code and locate where the MD5 hash is being calculated. They will need to look at the logic below and replace the current MD5 hash calculation with the below SHA256 HMAC calculation.

**Step 1: Create string with transaction data via the merchant application.**

Create a string with field names and values using the POST data from JSP\_VPC\_3Party\_Order.html:

|  |
| --- |
| String SHAhashAllFields(Map fields) {  hashKeys = "";  hashValues = "";  // create a list and sort it  List fieldNames = new ArrayList(fields.keySet());  Collections.sort(fieldNames);  // create a buffer for the SHA256 input  StringBuffer buf = new StringBuffer();  // iterate through the list and add the remaining field values  Iterator itr = fieldNames.iterator();  while (itr.hasNext()) {  String fieldName = (String) itr.next();  String fieldValue = (String) fields.get(fieldName);  hashKeys += fieldName + ", ";  if ((fieldValue != null) && (fieldValue.length() > 0)) {  buf.append(fieldName + "=" + fieldValue);  if (itr.hasNext()) {  buf.append('&');  }  }  } |

**Step 2: Create SHA256 HMAC**

* *Obtain Secure Hash Secret either from the existing MD5 calculation or from Merchant Administration*
* *Calculate SHA256 HMAC using the secret as a key to produce the secure hash*
* *Add the hash to the request string to send to the gateway*

|  |
| --- |
| byte[] mac = null;  try {  byte [] b = new BigInteger(SECURE\_SECRET, 16).toByteArray();  SecretKey key = new SecretKeySpec(b, "HmacSHA256");  Mac m = Mac.getInstance("HmacSHA256");  m.init(key);  m.update(buf.toString().getBytes("ISO-8859-1"));  mac = m.doFinal();  } catch(Exception e) {  }  String hashValue = hex(mac);  return hashValue;  }  .  .  .  // Create SHA256 secure hash and insert it into the hash map if it was created  // created. Remember if SECURE\_SECRET = "" it will not be created  if (SECURE\_SECRET != null && SECURE\_SECRET.length() > 0) {  String secureHash = SHAhashAllFields(fields);  fields.put("vpc\_SecureHash", secureHash);  fields.put("vpc\_SecureHashType", "SHA256");  } |

## Can I process 2.5 Party (Server Hosted with card details) transactions?

Yes, the process will use the same logic as above 3 Party example. Your developer will need to calculate the SHA256 HMAC and POST the data to the Payment Server.