BSG OrderBridge™

Integration Guide, v0.10

Daniel Duerr, Simon Welter, Gary Avery

ABSTRACT

Developer's guide to integration between merchant site and the BSG's OrderBridge payment platform.

Table of Contents

Section 1: Developer Introduction 3
- Checkout/Payment 3
- Merchant Callback 3
- Web Services API 3
- Message Authentication Code (MAC) 3

Section 2: Checkout/Payment Integration 5
- Subscriptions 5
- Example merchant checkout URLs 5
- Customer Information parameters 5
- Order Header parameters 6
- Order Line Item parameters 6
- Extensible Order Metadata parameters 6
- Subscription Header parameters 7
- Subscription Line Item parameters 8
- Extensible Subscription Metadata parameters 8
- Additional parameters 8
- Subscription Trial Periods 8
- Subscription Payment Processing 8
- Additional Subscription Processing Capabilities 9
- Message Authentication Code (MAC) parameters 9
- HTML Code Example 10
- Message Authentication Code (MAC) Example 10
- Test Payment Data 10

Section 3: Merchant Callback Integration 11
- Callback parameters 11
- Callback message types 11
- Timing of the callback during End User Checkout 11
- Callback Failure 12
- Message Authentication Code (MAC) parameters 12

Section 4: Web Services API Integration 13
- API URLs 13
- Common API request parameters 13
- Error response parameters 13
- Message Authentication Code (MAC) parameters 13
- Action: ‘credit’ 13
- Action: ‘process_subscription’ 14
- Action: ‘suspend_subscription’, ‘resume_subscription’, and ‘cancel_subscription’ 14
- Action: ‘get_subscription’ 14
- Action: ‘insert_order’ and ‘update_order’ 15
- Action: ‘get_orders’ 16
- Action: ‘get_order’ and ‘get_order_by_txn_id’ 16
- Action: ‘get_callbacks’ 16

Section 1: Developer Introduction

This guide describes the web development required to submit an order for payment processing via the BSG OrderBridge™ payment platform directly from a web site or e-commerce application. The OrderBridge was designed to drastically reduce the complexity, cost, and time to market traditionally associated with integrating an internet payment gateway.

Checkout/Payment

The OrderBridge provides a web-based payment and order storage platform that givese-commerce merchants the ability to quickly and easily integrate payment processing from BSG with an existing web site or e-commerce application. Unlike traditional e-commerce payment gateways that require low-level API integration, the OrderBridge payment process can be integrated by way of a simple HTML form submission from the merchant's web site.

For merchants who don't have an existing e-commerce system with order tracking, fulfillment, and reporting, the OrderBridge can also act as a order storage and reporting database. The OrderBridge admin portal enables the merchant to login and view/search their order history, perform basic transaction management (e.g.: refunds/credits), and generate reports.

Merchant Callback

The OrderBridge provides an HTTP-POST verification "callback" to a designated merchant URL which can notify the merchant web site to the outcome of an end user's checkout/payment process, enabling the merchant to confidently release the order to the end user (e.g.: digital goods, order fulfillment, etc.).

Web Services API

Finally, the OrderBridge provides a simple web services API that enables merchants to perform a variety of additional tasks via HTTP-POST, including:

- **Order Processing** – credit a previous payment transaction.
- **Subscriptions** – process a recurring subscription payment manually, update a subscription, cancel a subscription, or query a subscription.
- **Offline Order Storage** – store, update an order which was paid offline or by a 3rd party payment method (e.g.: check/money order, PayPal).
- **Reporting/Analytics** – retrieve a list of merchant orders matching various search criteria, a specific order by merchant order ID, or a specific order by payment transaction ID.

Message Authentication Code (MAC)

Allrequests sent from the merchant to the OrderBridge servers should include a signature parameter to verify the authenticity of the message. This signature serves as a Message Authentication Code (MAC) which verifies the authenticity of the merchant sending the message and that the message was not tampered with in

transit. The signature will also be included on all Merchant Callbacks sent from the OrderBridge to the merchant.

The signature can be generated by combining parameter values of the request in alphabetical order by parameter name along with the merchant's authkey (provided to the merchant by BSG), and then hashing this string using anMD5 hash.Assuming that the parameters for the OrderBridgerequestare as follows:

code=merchant,action=get_order, return=json, merchant_order_id=808

And assuming that the merchant's authkey is "12345678", a signature can be generated using the following PHP code:

```
<?php
// private key known only to merchant and OrderBridge
$authkey = "12345678";

// parameters and values
$code = "merchant";
$action = "get_order";
$return = "json";
$merchant_order_id = "808";

// concatenate the parameter values in alphabetical order by parameter name
$plain = $action.$code.$merchant_order_id.$return;

// then add the private authkey
$plain.= $authkey;

// now generate the signature using the md5() function
$signature = md5($plain);
?>
```

BSG may be able to provide code to generate the signature in additional languages upon request.

SECURITY WARNING: BSG highly recommends **not** creating the hash in the browser (i.e. JavaScript) as this leaves the merchant's authkey vulnerable and exposed to security threats. Instead, the hash should be generated by a server-side script where the authkey is only known behind that script.

IMPORTANT NOTE: Due to the large number of parameters being passed when sending a checkout POST to OrderBridge, we have a special list of parameters that must be used for generating the MAC signature. This list of parameters can be found on page 9 of this document.

Section 2: Checkout/Payment Integration

Integration with the OrderBridge for checkout/payment processing is achieved through an HTML form submission (HTTP-POST) from the merchant's web site to the OrderBridge merchant checkout URL, resulting in the end user being "redirected" the merchant's co-branded checkout/payment interface within the OrderBridge. From there, the end user can complete or cancel the order. In either event, the end user is redirected back to the merchant's web site upon concluding their OrderBridge checkout, delivering a seamless user interface flow, despite the redirection to and from the OrderBridge.

Subscriptions

In addition to an initial one-time purchase, the OrderBridge checkout interface supports the creation of an optional recurring subscription scheme, either in addition to, or in place of, the initial one-time order/purchase. If a recurring subscription scheme is desired, the following parameters should be passed as part of the HTTP-POST to the checkout interface. Setup of a subscription on its own will not create an order or collect any funds at the time of checkout. Instead, the subscription is meant to schedule a future order to be processed.

If an immediate up-front payment is desired as part of a subscription then an order must be submitted for the initial purchase along with a subscription for the future purchases, for example: if the end user is signing up for a monthly subscription with an initial payment upon checkout, you would pass an order for the initial purchase and a subscription with a *sub[start_date]* one month in the future for the future purchases.

If, however, the up-front payment is not required, then you can send a subscription by itself. In this case, the OrderBridge will authorize the first month's payment amount against the end user's method of payment, but it will wait until the *sub[start_date]* to process the first actual sale.

IMPORTANT NOTE: All Decimal type fields should be formatted as N.NN (i.e. two digits after the decimal point).

Example merchant checkout URLs

Item	URL
Staging	https://test.ob.bsgpayment.com/merchant/checkout/
Production	https://ob.bsgpayment.com/merchant/checkout/

Please substitute "merchant" in the URLs above with your unique merchant code.

Customer Information parameters

Name	Value	Format	Notes
customer[email]	Customer's email address	Up to 50 chars	Must be a valid email address
customer[first_name]	Customer's first name	Up to 32 chars	
customer[last_name]	Customer's last name	Up to 32 chars	
customer[phone]	Customer's phone number	Up to 15 chars	Optional

Order Header parameters

Name	Value	Format	Notes
order[merchant_order_id]	Merchant's Order ID	Up to 255 chars	Must be unique
order[billto_company]	Customer's "bill to" company	Up to 32 chars	Optional
order[billto_first_name]	Customer's "bill to" first name	Up to 32 chars	
order[billto_last_name]	Customer's "bill to" last name	Up to 32 chars	
order[billto_address1]	Customer's "bill to" address line 1	Up to 32 chars	
order[billto_address2]	Customer's "bill to" address line 2	Up to 32 chars	Optional
order[billto_city]	Customer's "bill to" city	Up to 32 chars	
order[billto_state]	Customer's "bill to" state	2 chars	Must be valid 2 char state code within country
order[billto_country]	Customer's "bill to" country	2 chars	Must be valid 2 char ISO country code
order[billto_zipcode]	Customer's "bill to" ZIP/postal code	Up to 10 chars	
order[shipto_company]	Customer's "ship to" company	Up to 32 chars	Optional
order[shipto_first_name]	Customer's "ship to" first name	Up to 32 chars	Optional
order[shipto_last_name]	Customer's "ship to" last name	Up to 32 chars	Optional
order[shipto_address1]	Customer's "ship to" address line 1	Up to 32 chars	Optional
order[shipto_address2]	Customer's "ship to" address line 2	Up to 32 chars	Optional
order[shipto_city]	Customer's "ship to" city	Up to 32 chars	Optional
order[shipto_state]	Customer's "ship to" state	2 chars	Optional, Must be valid 2 char state code within country
order[shipto_country]	Customer's "ship to" country	2 chars	Optional, Must be valid 2 char ISO country code
order[shipto_zipcode]	Customer's "ship to" ZIP/postal code	Up to 10 chars	Optional
order[total_subtotal]	Line item subtotal	Decimal	
order[total_discount]	Total discounts	Decimal	Optional
order[total_shipping]	Total shipping/handling	Decimal	Optional
order[total_tax]	Total tax	Decimal	Optional, Must match sum of order_item[N][tax]
order[total]	Grand total	Decimal	
order[ship_method]	Shipping/delivery method	Up to 20 chars	Optional

Order Line Item parameters

Name	Value	Format	Notes
order_item[N][sku]	Item SKU/number	Up to 25 chars	
order_item[N][name]	Item name	Up to 32 chars	
order_item[N][price]	Item price (each)	Decimal	
order_item[N][tax]	Item tax (each)	Decimal	Optional, Sum must match order[total_tax]
order_item[N][qty]	Item quantity	Integer	
order_item[N][description]	Item description	Up to 64kb	Optional
order_item[N][details][]	Item details	Up to 255 chars	Optional, Supports multiple values per line item, result is serialized into an array

Please substitute "N" in the parameter names with an incremental line item number (e.g.: 0, 1, 2, 3, etc.).

Extensible Order Metadata parameters

Name	Value	Format	Notes

order_metadata[key]	Value	Up to 255 chars	Optional

Please substitute "key" in the parameter name for the unique key (e.g.: "color", etc.).

Subscription Header parameters

Name	Value	Format	Notes
sub[merchant_subscription_id]	Merchant's Subscription ID	Up to 255 chars	Must be unique
sub[interval_length]	Number of time between Subscription Payments	Integer	
sub[interval_unit]	Frequency of Subscription Payments: 'days', 'weeks', 'months', 'years'	Fixed values	
sub[start_date]	Date to process first Subscription Payment	Date	Must be at least 1 day in the future
sub[total_occurrences]	Total number of Subscription Payments	Integer	Specify '9999' for unlimited
sub[trial_occurrences]	Number of occurrences in the Trial Period (if any)	Integer	Optional, See 'Subscription Trial Periods' below for more info
sub[trial_amount]	Total amount to bill during the Trial Period (if any)	Decimal	Optional, See 'Subscription Trial Periods' below for more info
sub[auto_process]	Enable Automatic Payment processing on OrderBridge?	Boolean	Optional, See 'Subscription Payment Processing' below for more info
sub[billto_company]	Customer's "bill to" company	Up to 32 chars	Optional
sub[billto_first_name]	Customer's "bill to" first name	Up to 32 chars	
sub[billto_last_name]	Customer's "bill to" last name	Up to 32 chars	
sub[billto_address1]	Customer's "bill to" address line 1	Up to 32 chars	
sub[billto_address2]	Customer's "bill to" address line 2	Up to 32 chars	Optional
sub[billto_city]	Customer's "bill to" city	Up to 32 chars	
sub[billto_state]	Customer's "bill to" state	2 chars	Must be valid 2 char state code within country
sub[billto_country]	Customer's "bill to" country	2 chars	Must be valid 2 char ISO country code
sub[billto_zipcode]	Customer's "bill to" ZIP/postal code	Up to 10 chars	
sub[shipto_company]	Customer's "ship to" company	Up to 32 chars	Optional
sub[shipto_first_name]	Customer's "ship to" first name	Up to 32 chars	Optional
sub[shipto_last_name]	Customer's "ship to" last name	Up to 32 chars	Optional
sub[shipto_address1]	Customer's "ship to" address line 1	Up to 32 chars	Optional
sub[shipto_address2]	Customer's "ship to" address line 2	Up to 32 chars	Optional
sub[shipto_city]	Customer's "ship to" city	Up to 32 chars	Optional
sub[shipto_state]	Customer's "ship to" state	2 chars	Optional, Must be valid 2 char state code within country
sub[shipto_country]	Customer's "ship to" country	2 chars	Optional, Must be valid 2 char ISO country code

sub[shipto_zipcode]	Customer's "ship to" ZIP/postal code	Up to 10 chars	Optional
sub[total_subtotal]	Line item subtotal	Decimal	
sub[total_discount]	Total discounts	Decimal	Optional
sub[total_shipping]	Total shipping/handling	Decimal	Optional
sub[total_tax]	Total tax	Decimal	Optional
sub[total]	Grand total	Decimal	
sub[ship_method]	Shipping/delivery method	Up to 20 chars	Optional

Subscription Line Item parameters

Name	Value	Format	Notes
sub_item[N][sku]	Item SKU/number	Up to 25 chars	
sub_item[N][name]	Item name	Up to 32 chars	
sub_item[N][price]	Item price (each)	Decimal	
sub_item[N][qty]	Item quantity	Integer	
sub_item[N][description]	Item description	Up to 64kb	Optional
sub_item[N][details][]	Item details	Up to 255 chars	Optional, Supports multiple values per line item, result is serialized into an array

Please substitute "N" in the parameter names with an incremental line item number (e.g.: 0, 1, 2, 3, etc.).

Extensible Subscription Metadata parameters

Name	Value	Format	Notes
sub_metadata[key]	Value	Up to 255 chars	Optional

Please substitute "key" in the parameter name for the unique key (e.g.: "color", etc.).

Additional parameters

Name	Value	Format	Notes
return_url	URL to return the end user to after successful payment	Up to 255 chars	Optional, defaults to the HTTP referrer
cancel_url	URL to return the end user to when they 'Cancel' payment	Up to 255 chars	Optional, defaults to the HTTP referrer
mac	Message Authentication Code (MAC)	32 char hex	See below

Subscription Trial Periods

If the *sub[trial_occurrences]* and *sub[trial_amount]* parameters are specified, a "trial" period will be enabled for the subscription. For each subscription payment processed in the trial, the difference between the *sub[total]* and *sub[trial_amount]* will be added to the *sub[total_discount]* amount of the resulting order. The *sub[trial_occurrences]* must be non-zero and less than *sub[total_occurrences]*, and the *sub[trial_amount]* must be non-zero and less than *sub*[*total*].

Subscription Payment Processing

If the *sub[auto_process]* parameter is set to *true*, the OrderBridge will automatically attempt to process the subscription payment on its due date. If approved, a new order will be created for the subscription payment and an email receipt will be sent to the end user. If declined or failed, the subscription process will be retried at regular intervals until the maximum number of "grace period" days has passed (e.g.: retry every 3 days until 10 days has passed). If the grace period has passed and the process has still not succeeded, the subscription will be suspended. In all of these

cases, the OrderBridge will send an appropriate Merchant Callback communicating the outcome of the process attempt so that the merchant's web site can decide what to do (e.g.: customer notification, suspension of access, etc.). For more information on Merchant Callbacks, please see Section 3.

If the *sub[auto_process]* parameter is set to *false*, the OrderBridge will not perform any automatic processing of the subscription payments. Instead, the merchant's web site will handle all of the automatic processing tasks and will send a *process_subscription*request to the OrderBridge Web Services API on the subscription's due date. If approved, a new order will be created for the subscription payment, an email receipt will be sent to the end user, and relevant information will be returned. If declined or failed, a descriptive error message will be returned. All retry and "grace period" logic will be performed by the merchant's web site. If/when the subscription has been declined or failed to the point of being considered uncollectable, the merchant's web site will send a *subscription_suspend* request to the API. The OrderBridge will NOT send any Merchant Callbacks when automatic processing has been disabled as the direct responses to the API requests will provide the same information in a more direct fashion.

Additional Subscription Processing Capabilities

The OrderBridge Web Services API provides a complete set of subscription processing "actions" which are described in Section 4 of this document. If you're considering the integration of the subscription capabilities, please review that section as well.

Message Authentication Code (MAC) parameters

The following parameters should be used in these sequences to generate the MAC signature:

Checkout Type	MAC parameters (in sequence)
Order only	code + customer[email] + order[merchant_order_id] + order[total] + order[total_subtotal] + authkey
Subscription only	code + customer[email] + sub[merchant_subscription_id] + sub[total] + sub[total_occurrences] + sub[total_subtotal] + sub[trial_amount] + sub[trial_occurrences] + authkey
Order + Subscription	code + customer[email] + order[merchant_order_id] + sub[merchant_subscription_id] + order[total] + order[total_subtotal] + sub[total] + sub[total_occurrences] + sub[total_subtotal] + sub[trial_amount] + sub[trial_occurrences] + authkey

HTML Code Example

The following example HTML would submit an order with one line item:

```
<form action="https://ob.bsgpayment.com/merchant/checkout/" method="post">
<input type="hidden" name="customer[email]" value="carissa@pycroft.com" />
<input type="hidden" name="customer[first_name]" value="Carissa" />
<input type="hidden" name="customer[last_name]" value="Pycroft" />
<input type="hidden" name="customer[phone]" value="(831) 531-2270" />
<input type="hidden" name="order[merchant_order_id]" value="100000546" />
<input type="hidden" name="order[billto_company]" value="Merchant, Inc." />
<input type="hidden" name="order[billto_first_name]" value="Carissa" />
<input type="hidden" name="order[billto_last_name]" value="Pycroft" />
<input type="hidden" name="order[billto_address1]" value="1861 Green Acres Rd" />
<input type="hidden" name="order[billto_city]" value="Madison" />
<input type="hidden" name="order[billto_state]" value="CT" />
<input type="hidden" name="order[billto_country]" value="US" />
<input type="hidden" name="order[billto_zipcode]" value="06443" />
<input type="hidden" name="order[total_subtotal]" value="19.99" />
<input type="hidden" name="order[total_tax]" value="2.00" />
<input type="hidden" name="order[total]" value="21.99" />
<input type="hidden" name="order_item[0][sku]" value="FOO" />
<input type="hidden" name="order_item[0][name]" value="Foo Item" />
<input type="hidden" name="order_item[0][price]" value="19.99" />
<input type="hidden" name="order_item[0][qty]" value="1" />
<input type="hidden" name="return_url" value="https://www.mystore.com/order/success/" />
<input type="hidden" name="cancel_url" value="https://www.mystore.com/order/cancel/" />
<p><button type="submit">Submit Order</button></p>
</form>
```

Message Authentication Code (MAC) Example

The following example PHP code would generate a MAC signature for the order in the HTML code example above:

```
<?php
// private key known only to merchant and OrderBridge
$authkey = "12345678";

// parameters and values
$code = "merchant";
$email = "carissa@pycroft.com";
$merchant_order_id = "100000546";
$order_total_subtotal = "19.99";
$order_total = "21.99";

// concatenate the parameter values in alphabetical order by parameter name
$plain = $code.$email.$merchant_order_id.$order_total.$order_total_subtotal;

// then add the private authkey
$plain.= $authkey;

// now generate the signature using the md5() function
$signature = md5($plain);
?>
```

Test Payment Data

The following test accounts can be used to achieve an 'Approval' for testing on the Staging URL:

Account Number	Additional Data Required for Approval
Credit Card: 4111111111111111	CVV: 123
Bill2Phone: 8885550000	Last 4 SSN: 1234

Section 3: Merchant Callback Integration

The merchant "callback" is a process by which the OrderBridge can notify the merchant's web site of information specific to an order. When a qualifying event occurs relative to an OrderBridge order, an HTTP-POST request is made to a specified URL within the merchant's web site. This process enables the OrderBridge to communicate order status information back to the merchant's web site in a secure and authenticated manner.

While the callback is not required for OrderBridge payment integration, it is **highly recommended** for accurate order processing. For security reasons, the callback URL must be defined within the merchant's OrderBridge profile – it cannot be passed as an HTTP-POST argument to the Checkout interface. BSG technical staff can assist with setup of the merchant profile.

Callback parameters

Name	Value	Format	Notes
message_type	Type of event	Fixed values (see below)	Type of event that occurred relative to the order and/or subscription and/or transaction
message_description	Description of the event	Up to 255 chars	
merchant_order_id	Merchant's Order ID	Up to 255 chars	Non-blank if event is order specific
merchant_subscription_id	Merchant's Subscription ID	Up to 255 chars	Non-blank if event is subscription specific
txn_id	Payment Transaction ID	Up to 32 chars	Non-blank if event is transaction specific
mac	Message Authentication Code (MAC)	32 char hex	See below

Callback message types

The following message types are generated by the OrderBridge callback system:

1. *CHECKOUT_CANCELLED*: generated when an end-user checkout is cancelled.
2. *CHECKOUT_COMPLETED*: generated when an end-user checkout is successfully completed.
3. *SUB_PROCESSED*: generated when an auto-processed subscription order/payment is processed.
4. *SUB_COMPLETED*: generated when an auto-processed subscription has processed its final order/payment.
5. *SUB_DECLINED*: generated when an auto-processed subscription order/payment is declined.
6. *SUB_FAILED*: generated when an auto-processed subscription order/payment is failed.
7. *SUB_CANCELLED*: generated when an auto-processed subscription is cancelled after too many retries.
8. *ORDER_CHARGEBACK*: generated when a chargeback is received against a previous order transaction.

Timing of the callback during End User Checkout

The callback request will always be made prior to the end user being shown the order completion/receipt page within the OrderBridge checkout, let alone the "Return to Merchant" button. This will enable the merchant to reliably complete its order processing routines prior to the end user's return (e.g.: release of digital goods, order fulfillment, etc.). In fact, the callback will enable the merchant complete these same routines even if end user fails to click the "Return to Merchant" button after completing the payment.

Callback Failure

If the callback attempt fails, the details of the callback message will be sent via email to the merchant's "webmaster" email address as defined in the merchant's OrderBridge profile. BSG technical staff can assist with setup of the merchant profile.

Message Authentication Code (MAC) parameters

The following callbackresponse parameters should be used to generate the MAC signature:

1. Merchant *code*
2. Callback *message_type*
3. Callback *merchant_order_id*
4. Callback *merchant_subscription_id*
5. Callback *txn_id*

Section 4: Web Services API Integration

Integration with the OrderBridge Web Services API is achieved by making an HTTP-POST request from the merchant's web site to the appropriate API. The API implements various "actions" or functions that it can perform. With each "action" come one or more required action-specific API parameters (in addition to the 'Common API request parameters' below). Upon completion of the requested action, the API will return a structured response array containing one or more parameters relevant to the action taken.

Because requests to the API are designed to return a response result directly, a separate merchant callback will not be generated for these API calls.

API URLs

Item	URL
Staging	https://test.ob.bsgpayment.com/srv/txn.php
Production	https://ob.bsgpayment.com/srv/txn.php

Common API request parameters

Name	Value	Format	Notes
code	Merchant's code	Up to 100 chars	
return	'json','serialize', or 'read'	Fixed values	Format to encode the structured response in
action	API "action" to call	Up to 32 chars	See descriptions of the various actions below
mac	Message Authentication Code (MAC)	32 char hex	See below

Error response parameters

Name	Value	Format	Notes
success	'false'	Boolean	
message	Error message/description	Text	
fields	One or more offending API parameter names	Array	

Message Authentication Code (MAC) parameters

For each of the following API actions, the Merchant *code*, API *action*, and all requiredaction-specific parameters should be used to generate the MAC signature, unless a specific action specifies otherwise.

Action: 'credit'

Performs a credit/refundofa previous 'sale' transaction.

Additional request parameters:

Name	Value	Format	Notes
merchant_order_id	Merchant's Order ID	Up to 255 chars	
txn_id	Payment Transaction ID of the original 'sale'	Up to 32 chars	
amount	Amount to credit	Decimal	Optional, defaults to original transaction amount

Success response parameters:

Name	Value	Format	Notes
success	'true'	Boolean	
status	'Approved'	Fixed values	
txn_id	Payment Transaction ID	Up to 32 chars	

Action: 'process_subscription'

Performs a subscription payment process against an *active*, *manually processed* subscription *with a payment due* to process. This action may be called as soon as one day prior to the payment due date, and anytime thereafter.

Additional request parameters:

Name	Value	Format	Notes
merchant_subscription_id	Merchant's Subscription ID	Up to 255 chars	
merchant_order_id	Merchant's Order ID to assign to the resulting order	Up to 255 chars	Optional, will be auto-generated if omitted

Success response parameters:

Name	Value	Format	Notes
success	'true'	Boolean	
merchant_order_id	Merchant's Order ID of the resulting order	Up to 255 chars	
txn_id	Payment Transaction ID of the resulting payment	Up to 32 chars	
amount	Amount of resulting payment	Decimal	

Action: 'suspend_subscription', 'resume_subscription', and 'cancel_subscription'

Updates the subscription status to the appropriate value.

Additional request parameters:

Name	Value	Format	Notes
merchant_subscription_id	Merchant's Subscription ID	Up to 255 chars	

Success response parameters:

Name	Value	Format	Notes
success	'true'	Boolean	

Action: 'get_subscription'

Retrieves the entire subscription object, including order information, the number of payment occurrences run, and the next payment due date.

Additional request parameters:

Name	Value	Format	Notes
merchant_order_id	Merchant's Order ID	Up to 255 chars	

Success response parameters:

Name	Value	Format	Notes

success	'true'	Boolean	
subscription	Full subscription data object	Object	

Action: 'insert_order' and 'update_order'

Inserts or updates an order which was processed outside of the OrderBridge. Please refer to Section 2 for complete list of 'Customer Information', 'Order Header', and 'Order Line Item' parameters. If inserting a new order, the *order[]* and *order_item[]* parameter sections are required and the *order_txn[]* parameter section is optional. If updating an existing order, all parameter sections are optional.

The*order[]* and *order_item[]* parameter sections can be updated, but the *order_txn[]* parameter section can only insert a new transaction. For example, if you wanted to insert an order with two transactions (a sale, and a credit) you would first call 'insert_order' with the full *order[]* and *order_item[]* parameter sections and the *order_txn[]* parameter section representing the first 'sale' transaction. Then, you would call 'update_order' with the *order_txn[]* parameter section representing the second 'credit' transaction. The result of these two subsequent actions would be a single order containing the two transactions.

For these two API actions, the Merchant *code*, API *action*, and *merchant_order_id* should be used to generate the MAC signature.

Additional request parameters:

Name	Value	Format	Notes
merchant_order_id	Merchant's Order ID	Up to 255 chars	
customer[]	Customer Information parameter group		Please see full 'Customer Information Parameters' list in Section 2
order[]	Order Header parameter group		Please see full 'Order Header Parameters' list in Section 2
order[status]	'pending', 'paid', 'cancelled', or 'refunded'	Fixed values	Optional, defaults to 'pending' on insert
order_item[]	Order Line Item parameter group		Please see full 'Order Line Item Parameters' list in Section 2
order_txn[acct_type]	Payment Transaction account type (e.g.: Check, PayPal, etc.)	Up to 25 chars	
order_txn[txn_id]	Payment Transaction ID	Up to 32 chars	
order_txn[txn_action]	'sale', 'credit', or 'void'	Fixed values	
order_txn[txn_amount]	Payment Transaction amount	Decimal	

Success response parameters:

Name	Value	Format	Notes
success	'true'	Boolean	

Action: 'get_orders'

Retrieves a list of merchant's order IDs. This action will return all of the merchant's order IDs if all parameters are omitted.

Additional request parameters:

Name	Value	Format	Notes
created[start]	Starting order creation date/time	Date/Time	Optional
created[end]	Ending order creation date/time	Date/Time	Optional
updated[start]	Starting order update date/time	Date/Time	Optional
updated[end]	Ending order update date/time	Date/Time	Optional
merchant_subscription_id	Merchant's Subscription ID	Up to 255 chars	Optional
status	'pending', 'paid', 'cancelled', or 'refunded'	Fixed values	Optional
process	'true' for orders processed by OrderBridge, 'false' for orders processed outside (and inserted via the web service)	Boolean	Optional

Success response parameters:

Name	Value	Format	Notes
success	'true'	Boolean	
orders	One or more matching merchant_order_id values	Array	

Action: 'get_order' and 'get_order_by_txn_id'

Retrieves the entire order object, either by *merchant_order_id* or *txn_id*, depending on the action requested.

Additional request parameters:

Name	Value	Format	Notes
merchant_order_id	Merchant's Order ID	Up to 255 chars	
txn_id	Payment Transaction ID	Up to 32 chars	

Success response parameters:

Name	Value	Format	Notes
success	'true'	Boolean	
order	Full order data object	Object	

Action: 'get_callbacks'

Retrieves a list of merchant's callbacks previously sent. This action will return the most recent 20 callbacks if all parameters are omitted.

Additional request parameters:

Name	Value	Format	Notes
created[start]	Starting callback creation date/time	Date/Time	Optional

created[end]	Ending callback creation date/time	Date/Time	Optional
lastsent[start]	Starting last sent date/time	Date/Time	Optional
lastsent[end]	Ending last sent date/time	Date/Time	Optional
response_code	HTTP response code	Integer	Optional (e.g.: 200, 500, 404)
start	Starting record number	Integer	Optional, defaults to '0'
limit	Number of rows to return from starting record number	Integer	Optional, defaults to '20'

Success response parameters:

Name	Value	Format	Notes
success	'true'	Boolean	
totalCount	Total number of matching callback records	Integer	
results	One or more matching callback records	Array	