

Methodology

Query

Overview

While our online reporting interface allows merchants to quickly and easily retrieve detailed information about past transactions, a need for additional flexibility may be required. For example, a merchant may have custom accounting software that requires up-to-date information about the settlement status of all credit card transactions every day.

This document describes how developers can query our reporting engine directly to retrieve transaction reports in a machine readable format. Once the data has been retrieved, it can then be parsed and imported into a variety of software applications.

Communication

The communication protocol used to send messages to the Payment Gateway is through the HTTP protocol over an SSL connection (HTTPS). The format you must use is name/value pairs delimited by ampersand.

URL: <https://secure.payscout.com/api/query.php>

Example Query String: `username=demo&password=password&transaction_id=123456789`

You may use either GET or POST to pass your request to the Query API. The name/value pairs that are accepted by the Payment Gateway can be found in the 'Variables' section of this API.

Variables

Query

Variable Name	Description(Allowed Values)[Format]
username*	Merchant username.
password*	Merchant password.
condition	A combination of values listed below can be passed and should be separated by commas. For example, to retrieve all transactions pending settlement or complete, the following could be used: Example: <code>condition=pendingsettlement,complete</code> pending: 'Auth Only' transactions that are awaiting capture. pendingsettlement: This transaction is awaiting settlement. failed: This transaction has failed. canceled: This transaction has been voided. complete: This transaction has settled.

unknown: An unknown error was encountered while processing this transaction.

transaction_type	<p>Retrieves only transactions with the specified transaction type. Use one of the following to specify payment type:</p> <p>cc: A credit card transaction.</p> <p>ck: A check transaction.</p> <p>Retrieves only transactions specified action types. A combination of the values can be used and should be separated by commas. For example, to retrieve all transactions with credit or refund actions, use the following:</p> <p>Example: action_type=refund,credit</p> <p>sale: Sale transactions.</p> <p>refund: Refund transactions.</p> <p>credit: Credit transactions.</p> <p>auth: 'Auth Only' transactions.</p> <p>capture: Captured transactions.</p> <p>void: Voided transactions.</p> <p>return: Electronic Check (ACH) transactions that have returned before (return) or after settlement (late_return).</p>
action_type	<p>Specify a transaction ID or a comma separated list of transaction IDs to retrieve information on.</p>
transaction_id	<p>Retrieves only transactions with the specified Order ID.</p>
order_id	<p>Retrieves only transactions with the specified last name.</p>
last_name	<p>Retrieves only transactions with the specified billing email address.</p>
email	<p>Retrieves only transactions with the specified credit card number. You can use either the full number or the last 4 digits of the credit card number.</p>
cc_number	<p>Retrieves only transactions with the specified merchant defined field value. Replace the '#' with a field number (e.g. merchant_defined_field_12=value)</p>
merchant_defined_field_#	<p>Only transactions that have been modified on or after this date will be retrieved. Note that any actions performed on a transaction will cause the modified date to be updated.</p>
start_date	<p>Format: YYYYMMDDhhmmss</p> <p>Only transactions that have been modified on or before this date will be retrieved. Note that any actions performed on a transaction will cause the modified date to be updated.</p>
end_date	<p>Format: YYYYMMDDhhmmss</p>

report_type	<p>Allows customer vault information or a html receipt to be returned. If you would like to query the Customer Vault to view what customer information is stored in the Customer Vault, you must set the customer_vault variable.</p> <p>If you omit the customer_vault_id, the system will return all customers that are stored in the vault. If you include a customer_vault_id, it will return the customer record associated with that ID.</p> <p>Example: report_type=customer_vault&customer_vault_id=123456789</p> <p>receipt: Will return an html receipt for a specified transaction id.</p> <p>customer_vault: Set the Query API to return Customer Vault data.</p> <p>Retrieves only transactions processed using the specified mobile device. The device IDs for this parameter are available in the License Manager. Use 'any_mobile' to retrieve all mobile transactions.</p>
mobile_device_license	<p>A combination of the values can be used and should be separated by commas. Can not be used with 'mobile_device_nickname'.</p> <p>Example 1: mobile_device_license=D91AC56A-4242-3131-2323-2AE4AA6DB6EB</p> <p>Example 2: mobile_device_license=any_mobile</p> <p>Retrieves only transactions processed using mobile devices with the specified nickname.</p>
mobile_device_nickname	<p>The nicknames for this parameter are available in the License Manager. Can not be used with 'mobile_device_license'.</p> <p>Example (URL encoded): mobile_device_nickname=Jim's%20iPhone</p>
customer_vault_id	<p>Set a specific Customer Vault record. Should only be used when report_type=customer_vault.</p>

Sample Response Query

```

<nm_response>
  <transaction>
    <transaction_id>1816519681</transaction_id>
    <platform_id></platform_id>
    <transaction_type>cc</transaction_type>
    <condition>pendingsettlement</condition>
    <order_id></order_id>
    <authorization_code>123456</authorization_code>
    <ponumber></ponumber>
    <order_description></order_description>
    <first_name></first_name>
    <last_name></last_name>
    <address_1></address_1>
    <address_2></address_2>
    <company></company>
    <city></city>
    <state></state>
    <postal_code></postal_code>
    <country>US</country>
    <email></email>
  </transaction>
</nm_response>

```

<phone></phone>
<fax></fax>
<cell_phone></cell_phone>
<customertaxid></customertaxid>
<customerid></customerid>
<website></website>
<shipping_first_name></shipping_first_name>
<shipping_last_name></shipping_last_name>
<shipping_address_1></shipping_address_1>
<shipping_address_2></shipping_address_2>
<shipping_company></shipping_company>
<shipping_city></shipping_city>
<shipping_state></shipping_state>
<shipping_postal_code></shipping_postal_code>
<shipping_country></shipping_country>
<shipping_email></shipping_email>
<shipping_carrier></shipping_carrier>
<tracking_number></tracking_number>
<shipping_date></shipping_date>
<shipping>0.00</shipping>
<shipping_phone></shipping_phone>
<cc_number>4xxxxxxxxxxx1111</cc_number>
<cc_hash>f6c609e195d9d4c185dcc8ca662f0180</cc_hash>
<cc_exp>1213</cc_exp>
<cavv></cavv>
<cavv_result></cavv_result>
<xid></xid>
<avs_response></avs_response>
<csc_response></csc_response>
<cardholder_auth></cardholder_auth>
<cc_start_date></cc_start_date>
<cc_issue_number></cc_issue_number>
<check_account></check_account>
<check_hash></check_hash>
<check_aba></check_aba>
<check_name></check_name>
<account_holder_type></account_holder_type>
<account_type></account_type>
<sec_code></sec_code>
<drivers_license_number></drivers_license_number>
<drivers_license_state></drivers_license_state>
<drivers_license_dob></drivers_license_dob>
<social_security_number></social_security_number>
<processor_id>ccprocessora</processor_id>
<tax>0.00</tax>
<currency>USD</currency>
<surcharge>2.50</surcharge>
<tip></tip>
<cc_bin>411111</cc_bin>
<action>
 <amount>17.50</amount>
 <action_type>sale</action_type>
 <date>20130218204917</date>
 <success>1</success>
 <ip_address>50.76.64.233</ip_address>
 <source>mobile</source>
 <username>demo</username>
 <response_text>SUCCESS</response_text>
 <batch_id>0</batch_id>

```
<processor_batch_id></processor_batch_id>
<response_code>100</response_code>
<device_license_number>D91AC56A-4242-3131-2323-2AE4AA6DB6EB</device_
<device_nickname>Sam's iPhone</device_nickname>
</action>
</transaction>
</nm_response>
```