

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:	<code>https://centavo.transactiongateway.com/api/query.php</code>
Example Post Data:	<code>security_key=security_key&transaction_id=123456789</code>

You should POST 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.

The Query API can be tested with live credentials or a dedicated test account only. Please contact your Merchant Service Provider for more information.

The Query API will respond in Universal Time Coordinated (UTC).

Variables

Query

POST URL

POST URL: <https://centavo.transactiongateway.com/api/query.php>

Variable Name	Description(Allowed Values)[Format]
security_key*	API Security Key assigned to a merchant account. New keys can be generated from the merchant control panel in Settings > Security Keys
condition	<p>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:</p> <p>Example: condition=pendingsettlement,complete</p> <p><i>pending</i>: 'Auth Only' transactions that are awaiting capture.</p> <p><i>pendingsettlement</i>: This transaction is awaiting settlement.</p> <p><i>in_progress</i>: This Three-Step Redirect API transaction has not yet been completed. The transaction condition will change to 'abandoned' if 24 hours pass with no further action.</p> <p><i>abandoned</i>: This Three-Step Redirect API transaction has not been completed, and has timed out.</p> <p><i>failed</i>: This transaction has failed.</p> <p><i>canceled</i>: This transaction has been voided.</p> <p><i>complete</i>: This transaction has settled.</p> <p><i>unknown</i>: An unknown error was encountered while processing this transaction.</p>
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>action_type</p>	<p>Retrieves only transactions with the 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><i>sale</i>: Sale transactions.</p> <p><i>refund</i>: Refund transactions.</p> <p><i>credit</i>: Credit transactions.</p> <p><i>auth</i>: 'Auth Only' transactions.</p> <p><i>capture</i>: Captured transactions.</p> <p><i>void</i>: Voided transactions.</p> <p><i>return</i>: Electronic Check (ACH) transactions that have returned, as well as credit card chargebacks.</p>
<p>source</p>	<p>Retrieves only transactions with a particular 'transaction source'. A combination of the values can be used and should be separated by commas. For example, to retrieve all transactions with api or recurring actions, use the following:</p> <p>Example: source=api,recurring</p> <p><i>api</i>: API transactions.</p> <p><i>batch_upload</i>: Batch Upload transactions.</p> <p><i>mobile</i>: Mobile (iProcess) transactions.</p> <p><i>quickclick</i>: QuickClick transactions.</p> <p><i>quickbooks</i>: QuickBooks SyncPay transactions.</p> <p><i>recurring</i>: Recurring transactions when using Recurring module.</p> <p><i>swipe</i>: SwIPE Software transactions.</p> <p><i>virtual_terminal</i>: Virtual Terminal transactions.</p> <p><i>internal</i>: Internal transactions. Typically indicates settlement</p>
<p>transaction_id</p>	<p>Specify a transaction ID or a comma separated list of transaction IDs to retrieve information on. Alternatively, provide a Subscription ID to retrieve processed (approved and declined) transactions associated with it.</p>

subscription_id	Set a specific subscription record or comma separated list of records. Using this with a transaction search will return all transactions associated with this subscription. This will return this subscription's payment/plan information when used with report_type=recurring.
invoice_id	Set a specific Invoice ID. Should only be used when report_type=invoicing.
partial_payment_id	Retrieves only transactions with the specified partial payment ID.
order_id	Retrieves only transactions with the specified Order ID.
first_name	Retrieves only transactions with the specified first name.
last_name	Retrieves only transactions with the specified last name.
address1	Retrieves only transactions with the specified specified address.
city	Retrieves only transactions with the specified city.
state	Retrieves only transactions with the specified state.
zip	Retrieves only transactions with the specified zip/postal code.
phone	Retrieves only transactions with the specified phone number.
fax	Retrieves only transactions with the specified fax number.
order_description	Retrieves only transactions with the specified order description.
drivers_license_number	Retrieves only transactions with the specified driver's license number.
drivers_license_dob	Retrieves only transactions with the specified driver's license date of birth.
drivers_license_state	Retrieves only transactions with the specified driver's license state.
email	Retrieves only transactions with the specified billing email address.
cc_number	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.

merchant_defined_field_#	<p>Retrieves only transactions with the specified merchant defined field value. Replace the '#' with a field number (1-20) (Example: merchant_defined_field_12=value)</p>
start_date	<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> <p>Format: YYYYMMDDhhmmss</p>
end_date	<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> <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><i>receipt</i>: Will return an html receipt for a specified transaction id.</p> <p><i>customer_vault</i>: Set the Query API to return Customer Vault data.</p> <p><i>recurring</i>: Set the Query API to return subscription data.</p> <p><i>recurring_plans</i>: Set the Query API to return plan data.</p> <p><i>invoicing</i>: Set the Query API to return invoicing data.</p> <p><i>gateway_processors</i>: Will return Processor details a user has permissions for. Specify a "user" by querying with that security key.</p> <p><i>account_updater</i>: Will return Customer Vault data that has been updated using the Account Updater service.</p>

test_mode_status: Will return whether the account has test mode active or inactive.

mobile_device_license

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.
A combination of the values can be used and should be separated by commas.
Can not be used with 'mobile_device_nickname'.

Example 1:
mobile_device_license=D91AC56A-4242-3131-2323-2AE4AA6DB6EB
Example 2:
mobile_device_license=any_mobile

mobile_device_nickname

Retrieves only transactions processed using mobile devices with the specified nickname.
The nicknames for this parameter are available in the License Manager.
Can not be used with 'mobile_device_license'.

Example (URL encoded):
mobile_device_nickname=Jim's%20iPhone

customer_vault_id

Set a specific Customer Vault record. Should only be used when
report_type=customer_vault.

date_search

Allows Customer Vault information to be returned based on the 'created' or 'updated' date. If you would like to query the Customer Vault to view when customer information was created or updated, you must set the report_type variable with the customer_vault value.

If you omit the report_type variable, the system will ignore the date_search variable.

Example:
report_type=customer_vault&date_search=created,updated&start_date=20170101000000&end_date=20201231232359

created: Will return Customer Vault data created during a specified date range.

<i>updated</i> : Will return Customer Vault data updated during a specified date range.	
result_limit	Determines the maximum number of results that may return for the current query.
page_number	Determines which "page" of results are returned. For example, "result_limit=100&page_number=0" will return the first 100 results. Using "page_number=1" will return the next 100 results. Default is "0".
result_order	Determines order of in which results are returned. Default value is "standard".
	<i>standard</i> : Returns the results from oldest to newest.
	<i>reverse</i> : Returns the results from newest to oldest.
invoice_status	Specify a comma separated list of what invoice statuses will return when using an "invoicing" report type.
	Example: invoice_status=open,closed,paid
	<i>open</i> : Open Invoices
	<i>paid</i> : Paid Invoices
	<i>closed</i> : Closed Invoices
	<i>past_due</i> : Past Due Invoices

Sample Response

Query

```
<nm_response>
<transaction>
<transaction_id>2612675976</transaction_id>
<partial_payment_id></partial_payment_id>
<partial_payment_balance></partial_payment_balance>
<platform_id></platform_id>
<transaction_type>cc</transaction_type>
<condition>complete</condition>
<order_id>1234567890</order_id>
<authorization_code>123456</authorization_code>
<ponumber></ponumber>
<order_description></order_description>
<first_name>John</first_name>
<last_name>Smith</last_name>
<address_1>123 Main St</address_1>
<address_2>Apt B</address_2>
<company></company>
<city>New York City</city>
<state>NY</state>
<postal_code>10001</postal_code>
<country>US</country>
<email>johnsmith@example.com</email>
<phone>1234567890</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>1.00</shipping>
<shipping_phone></shipping_phone>
<cc_number>4xxxxxxxxxxx1111</cc_number>
<cc_hash>f6c609e195d9d4c185dcc8ca662f0180</cc_hash>
<cc_exp>1215</cc_exp>
<cavv></cavv>
<cavv_result></cavv_result>
<xid></xid>
<eci></eci>
<directory_server_id></directory_server_id>
<three_ds_version></three_ds_version>
<avs_response>N</avs_response>
```


<csc_response>M</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>processora</processor_id>
<tax>1.00</tax>
<currency>USD</currency>
<surcharge></surcharge>
<cash_discount></cash_discount>
<tip></tip>
<card_balance></card_balance>
<card_available_balance></card_available_balance>
<entry_mode>Keyed</entry_mode>
<cc_bin>411111</cc_bin>
<cc_type>visa</cc_type>
<signature_image>
iVBORw0KGgoAAAANSUUhEUgAAABAAAAAQCAIAAACQkWg2AAAAAXNSR0IArs4c6QAAAARnQU1
BAA
Cxjwv8YQUAAAAJcEhZcwAAEnQA
ABJ0Ad5mH3gAAACGSURBVDhPlZGBEoAgCE01//9nG83mxMrr3dkBG0hVW2vFqLX26CYbPic
7yS
AVe8LBq5u4elyV4M0NXIoGXyqA
w4QqMAwJCRu+Az7HslvgHtexlFKwGrLjG/h/rESmhrhRnLCKwjiNeLYv5QsXOoNSig8IsNY
aZ0
tXJoGU9hU1k18XlZLOQHTenf7I
cf3BwAAAABJRu5ErkJggg==
</signature_image>
<product>
<sku>RS-100</sku>
<quantity>1.0000</quantity>
<description>Red Shirt</description>
<amount>10.0000</amount>
</product>
<action>
<amount>11.00</amount>
<action_type>sale</action_type>
<date>20150312215205</date>
<success>1</success>
<ip_address>1.1.1.1</ip_address>
<source>virtual_terminal</source>
<api_method></api_method>
<username>demo</username>
<response_text>SUCCESS</response_text>
<batch_id>0</batch_id>
<processor_batch_id></processor_batch_id>
<response_code>100</response_code> <processor_response_text>NO
MATCH</processor_response_text>
<processor_response_code>00</processor_response_code>

```
<requested_amount>11.00</requested_amount>
<device_license_number></device_license_number>
<device_nickname></device_nickname>
</action>
<action>
<amount>11.00</amount>
<action_type>level3</action_type>
<date>20150312215205</date>
<success>1</success>
<ip_address>1.1.1.1</ip_address>
<source>virtual_terminal</source>
<api_method></api_method>
<username>demo</username>
<response_text></response_text>
<batch_id>0</batch_id>
<processor_batch_id></processor_batch_id>
<response_code>100</response_code>
<processor_response_text></processor_response_text>
<processor_response_code></processor_response_code>
<device_license_number></device_license_number>
<device_nickname></device_nickname>
</action>
<action>
<amount>11.00</amount>
<action_type>settle</action_type>
<date>20150313171503</date>
<success>1</success>
<ip_address></ip_address>
<source>internal</source>
<api_method></api_method>
<username></username>
<response_text>ACCEPTED</response_text>
<batch_id>76158269</batch_id>
<processor_batch_id>782</processor_batch_id>
<response_code>100</response_code>
<processor_response_text></processor_response_text>
<processor_response_code>000000000021980</processor_response_code>
<device_license_number></device_license_number>
<device_nickname></device_nickname>
</action>
</transaction>
</nm_response>
```