CardQuiry, Inc.
Balance Inquiry API
Version 1.5

1. CardQuiry Overview

This document describes the CardQuiry, Inc. (CardQuiry) Balance Inquiry API. This REST message specification describes the transaction set currently supported by CardQuiry and is intended to be used / referenced by:

- Technical Project Managers
- Technical Business Analysts
- Technical Architects
- Engineering Resources
- Developers
- Technical Support resources

2. CardQuiry Services

The CardQuiry REST API allows access to web services as URI-addressable resources. Because these resources can be accessed through HTTP method requests, the interaction with resources is system independent. Virtually any HTTP client application can be used to access the following CardQuiry services:

1. The giftcardbalance standard service provides information about the card number and PIN (if applicable) submitted by the client.

2. The giftcardbalance/{requestId} standard service supports any balance inquiry that requires a deferred response.

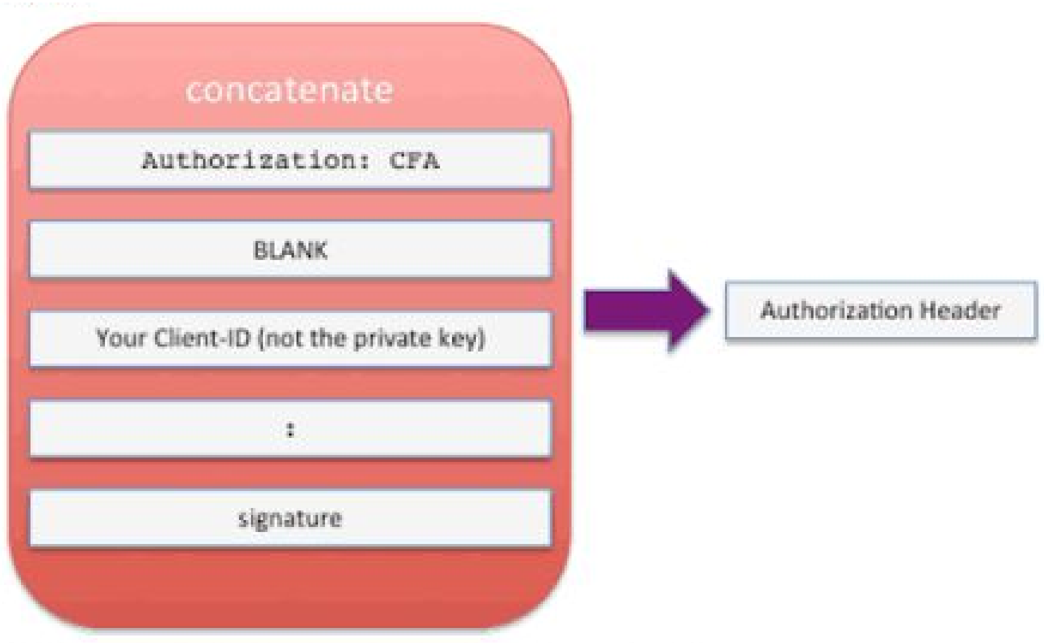
3. Using the REST API
To access the services, the client application initiates HTTP method requests (such as GET and POST) to the corresponding REST API resource. The CardQuiry server then processes the requests and returns appropriate responses in a JSON format.

3.1 REST Authentication

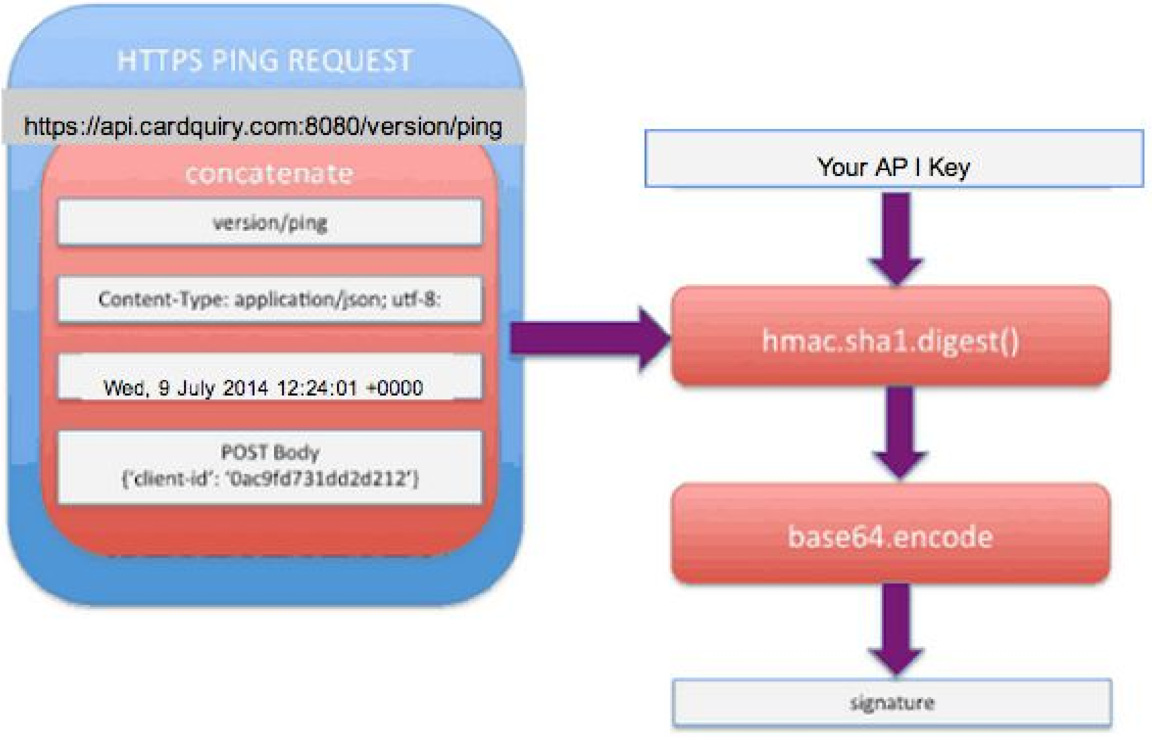
Every request to CardQuiry services must be digitally signed with your private key. The CardQuiry system validates each request with your public key. In addition, each CardQuiry response is also digitally signed. We strongly encourage all subscribers to validate the signature in all CardQuiry responses.

Signing your requests proceeds in three steps:

1. Calculating the signature



2. Incorporating the calculated signature and your customer ID in the request's authorization header.


HTTPS PING REQUEST
https://api.cardquiry.com:8080/version/ping
concatenate
version/ping
Content-Type: application/json; utf-8:
Wed, 9 July 2014 12:24:01 +0000
POST Body
{'client-id': '0ac9fd731dd2d212'}
Your AP I Key
hmac.sha1.digest()
base64.encode
signature

3. Add the authorization header to the header section of your HTTP request.

3.2 Requests

Get Balance API Request

POST https://api.cardquiry.com:8080/api/giftcardbalance

URL to initiate a balance request for a gift card. If the card balance cannot be immediately provided (due to a manual verification requirement) then the server will send a "010" responseCode and the caller should try again later using the deferred response URL described below.

Parameters	**Definition**	**Presence**	**Notes/Rules**
retailerID	VARCHAR(20)	M	See Appendix with Identification Information
cardNumber	VARCHAR(20)	M	
PIN	VARCHAR(20)	M	
Version	VARCHAR(20)	M	Use the version number to access newer features.

Parameters	**Definition**	**Presence**	**Notes/Rules**
RequestID	VARCHAR(20)	M	
Balance	DECIMAL(18,2)	M	
ResponseCode	VARCHAR(20)	M	See Appendix for full set of codes
ResponseMessage	VARCHAR(60)	M	See Appendix for full set of messages

VerificationType	VARCHAR(20)	M	See Appendix for details

Response Sample

Response sample for immediate balance check:

{“requestId”:”200000000”, “verificationType”:”PJVT_BOT”,”responseDateTime”:”2014-07-09 17:03:56.0”,”responseCode”:”000”,”responseMessage”:”success”,”balance”:12.35}

Get Balance API Deferred

GET https://api.cardquiry.com:8080/api/giftcardbalance/{requestID}
No parameters are required; same response as the POST request

Response Sample

Deferred Response Sample:

{“requestId”:”200000001”, “verificationType”:”PJVT_BOT”, ”responseCode”:”010”, ”responseMessage”:”Delayed verification required”, ”responseDateTime”:”2014-07-09 17:07:16.784”, ”recheckDateTime”:”2014-07-09 17:37:16.784”}

4. Appendix

4.1 *Response Code Table*

The table below defines the various data types that are being used in this document and their meaning:

Response Code	Response Message	Client Action
000	Success	N/A
010	Deferred Response	Wait for 000 or 179
179	Timeout	Manual Check
180	Header Data Error	Contact CQ

181	Authentication Error	Contact CQ
182	Unknown Request ID	Contact CQ
207	Invalid Retailer PAN	Verify Data
900011	Retailer Data Incorrect	Verify Data
900016	Retailer Not Supported	Manual Check
900020	In Store Balance Only	Obtain Balance in Store Only

4.2 Retailer Identification Information

A supplemental document will be provided by CardQuiry, Inc. at the time of implementation that contains a list of all retailer identification values. An example of a retailer identification number is as follows:

114

Note: CardQuiry continues to add new retailers. On average this list will be updated once per month. While developing a solution to support this dynamic list, please keep in mind that retailers will be added and, in some cases, removed over time.

4.3 Retailer Verification Type

Retailer responses are typically returned in near-real time. In some cases, there are retailers that will require some manual intervention to ensure an accurate balance is obtained. CardQuiry refers to those retailers as having a deferred response. Please see the verification type below for a deferred response.

Verification: PJVT_BOT

4.4 Fake Card Numbers

We have added fake card numbers to our system which will generate sample responses which can be used for testing especially while starting to integrate with us. The sample responses generated by these fake card numbers are similar to the response generated by a real card which has a similar response code. The fake card numbers are:

- **111111111111111** - returns success, response code =**'000'**
- **222222222222222** - returns deferral, response code = **'179'**

- **3333333333333333** - returns bad card, response code = **"900011'**

4.4 Retailer Site and Bot Status

The status of our balance inquiry bot and the retailer website can be found out at: https://api.cardquiry.com:8080/CheckBotStatus?retailerid=<retailerid>. The response messages for this url are the following jsons:

1. **{'RetailerID': '2', 'Site Status': 'up', 'Bot Status': 'down'}** - The current status of the retailer website and CQ bot for that retailer.
2. **{'Error':'Retailer not found'}** - The retailerid send is incorrect.

In order to send bot/site status updates to the client the response json send back to the each client after an update of bot status will have additional information on bot/site status with it. The additional fields are:

1. **Site_up_Bot_up** - List of retailerid's with a working bot and retailer website
2. **Site_up_Bot_down** - List of retailerid's with a troubled bot but a working retailer website
3. **Site_down_Bot_down** - List of retailerid's with a troubled bot and retailer website

An example response json right after update is is :

{"verificationType": "PJVT_BOT", "responseCode": "900011", "**Site_down_Bot_down**": "116", "**Site_up_Bot_up**": "10,100,1", "**Site_up_Bot_down**": "114,117", "requestId": "160051594907795773", "responseDateTime": "2014-10-13 04:50:25.736831", "balance": "Null", "responseMessage": "Retailer Data Incorrect"}.