



**SPARROW**

*Transact, Grow, SPARROW.*

## **Merchant Public Services**

Settlement API

**Version 1.3**

**Released** February 2017

## Revision History

Date	Revision	Comments	Author
2016-03-10	1.0	Initial document created	Vlad Volosuk
2016-03-23	1.1	Action route URL is corrected	Vlad Volosuk
2017-01-19	1.2	Streamline Authentication is added	Sergey Belov
2017-03-01	1.3	Review document	Ilya Tretyakov



## Overview

The Sparrow Gateway provides a set of services which can be used for integrating the gateway with merchant's IT systems. Merchant Public API provides the following RESTfull services:

- **Account** service – it is auxiliary service to obtain the authentication token required for invoking other Merchant Public API services
- **Transaction** service - this service allows querying transactions processed by the merchant in the Sparrow Gateway
- **Terminal** service – this service provides control functions over transaction processing

This document provides architectural overview of Merchant Public API and describes the services included into this API.

## Architectural Overview

Merchant Public API consists of a number of *Actions* which are logically grouped into services. This section describes the structure of HTTP requests which are used for invoking *Actions*.

The legend:

- **<TERM>** indicates a parameter that will be described below of the snippet
- **[any-text]** indicates an optional part that can be not provided in requests for some *Actions* (see the specification of the particular *Action*)

### Action Route

A route to an *Action* is a URL-based path that uniquely identifies the *Action* for invocation and invocation parameters:

```
<URL>/api/public/merchant/<SERVICE>/<ACTION> [ /<PATH-PARAMETERS> ] [ ?<QUERY-PARAMETERS> ]
```

Term	Description
<URL>	The address of the Sparrow Gateway server where the service is hosted.  Merchants have to use the following URL for the production (life) integrations:  <a href="https://secure.sparrowone.com">https://secure.sparrowone.com</a>
<SERVICE>	The name of the service where the <i>Action</i> is located. Examples: <b>account</b> , <b>transactions</b> , <b>terminal</b> . See documentation of services below in this document.
<ACTION>	The name of the <i>Action</i> . See documentation of services below in this document.  Note: normally the name of the action must be always presented in <i>Action Route</i> . However, the service can have a default Action. In this case, this action is invoked without the name in the <i>Action Route</i> . Example: <b>authenticate</b> action in the <b>account</b> service.

<b>&lt;PATH-PARAMETERS&gt;</b>	Path parameters are used to clarify an entity for which the action has to be invoked
<b>&lt;QUERY-PARAMETERS&gt;</b>	Additional parameters provided in the form of HTTP query string

## Action Request

An *Action* can be invoked by a HTTP request:

```

<HTTP-VERB> <ACTION-ROUTE> HTTP/1.1
Content-Type: <REQUEST-CONTENT-TYPE>
Accept: <RESPONSE-CONTENT-TYPE>
Authorization: <REQUEST-AUTH-PARAMS>
[other HTTP headers]

<REQUEST-DATA>
    
```

Term	Description
<b>&lt;HTTP-VERB&gt;</b>	<p>HTTP verb for invoking the service <i>Action</i>. Possible values are:</p> <ul style="list-style-type: none"> <li>▪ GET</li> <li>▪ POST</li> <li>▪ PUT</li> <li>▪ DELETE</li> </ul> <p>It is not mandatory for <i>Actions</i> to support all 4 verbs (but at least one verb must be supported). The verb can be interrupted as a sub-action supported by the <i>Action</i>. See the service documentation to select the proper verb for the <i>Action</i>.</p>
<b>&lt;ACTION-ROUTE&gt;</b>	An URL-based path that uniquely identifies the <i>Action</i> for invocation and invocation parameters. See the section above for details.
<b>&lt;REQUEST-CONTENT-TYPE&gt;</b>	<p>The content type of the <b>&lt;REQUEST-DATA&gt;</b>. Possible values are:</p> <ul style="list-style-type: none"> <li>▪ <i>application/xml</i></li> <li>▪ <i>application/json</i></li> </ul> <p>See <i>JSON Objects</i> and <i>XML Objects</i> sections below for object format details.</p>
<b>&lt;RESPONSE-CONTENT-TYPE&gt;</b>	<p>The expected content type for the response. Possible values are:</p> <ul style="list-style-type: none"> <li>▪ <i>application/xml</i></li> <li>▪ <i>application/json</i></li> </ul> <p>If <i>Accept</i> HTTP header is not provided in the request, then the response will be returned in JSON format.</p> <p>See <i>JSON Objects</i> and <i>XML Objects</i> sections below for object format details.</p>
<b>&lt;REQUEST-AUTH-PARAMS&gt;</b>	The authorization parameters. See <i>Authentication</i> section.
<b>&lt;REQUEST-DATA&gt;</b>	Request data object formatted according <b>&lt;REQUEST-CONTENT-TYPE&gt;</b> . Names of request parameters are case-sensitive.

	See the <i>Action</i> documentation for parameters supported by the <i>Action</i> in <b>&lt;REQUEST-DATA&gt;</b> .
--	--

## Action Response

Here is a format of *Action* responses:

```
HTTP/1.1 <HTTP-CODE> <HTTP-CODE-DESCRIPTION>
Content-Type: <RESPONSE-CONTENT-TYPE>
WWW-Authenticate: <RESPONSE-AUTH-TOKEN>
[other HTTP headers]

<RESPONSE-DATA>
```

Term	Description
<b>&lt;HTTP-CODE&gt;</b>	<p>A standard HTTP response code. Here are codes (with HTTP descriptions) commonly used in Merchant Public API:</p> <ul style="list-style-type: none"> <li>▪ 200 OK The action was executed successfully</li> <li>▪ 403 Forbidden The current user is not allowed to execute this <i>Action</i></li> <li>▪ 404 Not Found The Action was requested for the entity that is not found on the system</li> <li>▪ 400 Bad Request The Action was executed with an error. More information about the error is returned in <b>&lt;RESPONSE-DATA&gt;</b>.</li> <li>▪ 500 Internal Server Error It is unwanted, but possible response that indicates about internal (unexpected) error during processing the <i>Action</i>.</li> </ul> <p>Particular <i>Actions</i> may support other HTTP response codes, see the <i>Action</i> documentation for details.</p>
<b>&lt;HTTP-CODE-DESCRIPTION&gt;</b>	A general message that clarifies the <b>&lt;HTTP-CODE&gt;</b> . See above for examples.
<b>&lt;RESPONSE-CONTENT-TYPE&gt;</b>	<p>The content type of <b>&lt;RESPONSE-DATA&gt;</b>. Possible values are:</p> <ul style="list-style-type: none"> <li>▪ <i>application/xml</i> (see <i>XML Objects</i> section for details)</li> <li>▪ <i>application/json</i> (see <i>JSON Objects</i> section for details)</li> </ul> <p>See <i>JSON Objects</i> and <i>XML Objects</i> sections below for object format details.</p>
<b>&lt;RESPONSE-AUTH-TOKEN&gt;</b>	The authentication token. The response contains WWW-Authenticate HTTP header only when the request contains valid authentication parameters: credentials or API Key. See <i>Authentication</i> section.
<b>&lt;RESPONSE-DATA&gt;</b>	<p>Response parameters formatted according <b>&lt;REQUEST-CONTENT-TYPE&gt;</b>.</p> <p>Names of response parameters are case-sensitive.</p>

See the <i>Action</i> documentation for parameters supported by the <i>Action</i> in <b>&lt;REQUEST-DATA&gt;</b> .
--

### JSON Objects

Action requests and responses can be formatted as JSON objects:

```
{
  "Username": "JoeDoe",
  "Password": "pwd"
}
```

See *Action* documentation for supported request/response parameters.

### XML Objects

Action requests and responses can be formatted as XML objects:

```
<LoginResponse>
  <AuthToken>TIUPQV+R1R5RGORA4IZVUIXIYEP+OFNB</AuthToken>
</LoginResponse>
```

See *Action* documentation for supported request/response parameters.

The name of the root element is composed as shown below:

- The root element for *Action* requests is named **<ActionName>Request** (e.g. LoginRequest)
- The root element for *Action* responses is named **<ActionName>Response** (e.g. LoginResponse)

where **<ActionName>** is the name of the current *Action*, the first letter is the upper case, others – in the low case.

### Authentication

Only authenticated users are allowed to invoke service *Actions*. Merchant Public API supports two action invocation workflows which differ in a way how authentication information is obtained and specified in API calls: explicit authentication and streamline authentication.

#### Streamline Authentication

Streamline authentication is a preferable way to work with Merchant Public API: authentication information is specified with the every API call in HTTP headers as Authorization parameter (see **Action Request** section above). The following forms are supported for this parameter:

- **Authorization: credentials=<credentials-string>**  
where <credentials-string> is <username>:<password> encoded by BASE64

This form is used for the first API call if the user-based authentication scheme is used. Please note, the username cannot contain “:” symbol, otherwise the expression cannot be parsed correctly (the gateway’s policy does not allow using “:” in user names as well).

- **Authorization: apikey=<api-string>**  
This form is used for the first API call if the API-based authentication scheme is used.

- **Authorization: token=<auth-token>**  
**Authorization: <auth-token>**

These two forms are equal. They are recommended for using with second and forth API calls. <auth-token> is provided as WWW-Authenticate HTTP header in the response for the first API call: **WWW-Authenticate: <auth-token>** (see **Action Response** section above).

### Explicit Authentication

Explicit authentication can be used if an integration does not have ability to specify/fetch HTTP header parameters. This workflow assumes that a special API call is made just to obtain an auth token.

- The app invoke **Authenticate** action (see **Account Service** section below). If the action is successful, then <auth-toke> is returned in the response.
- The app makes the first API call with providing **token=<auth-token>** in the request’s query string
- The second and forth API calls are performed in the same way: by providing **token=<auth-token>** in the request’s query string

## Account Service

The **Account** service is auxiliary service that allows retrieving the authentication token required for invoking other *Actions*.

### Authenticate Action

This action is used to obtain the authentication token for the specified user.

Name	Value
<b>Service</b>	Account
<b>Action Name</b>	(default)
<b>Method</b>	POST
<b>Path Parameters</b>	No
<b>Query Parameters</b>	No
<b>Request</b>	The following items must be provided in <b>&lt;REQUEST-DATA&gt;</b> : <ul style="list-style-type: none"> <li>▪ <b>Username</b></li> <li>▪ <b>Password</b></li> </ul>
<b>Response</b>	The action returns the following items in <b>&lt;RESPONSE-DATA&gt;</b> : <ul style="list-style-type: none"> <li>▪ <b>AuthToken</b> – an authentication token for subsequent service actions calls</li> </ul>

## Terminal Service

The **Terminal** service provides a set of *Actions* over payment transactions.

**Account Settle Action**

This action is used to settle all transactions for the specified account which require settlement.

Name	Value
<b>Service</b>	Terminal
<b>Action Name</b>	Settle
<b>Method</b>	POST
<b>Path Parameters</b>	No
<b>Query Parameters</b>	The following items must be provided in <b>&lt;QUERY-PARAMETERS&gt;</b> : <ul style="list-style-type: none"> <li>▪ <b>token</b> – the authentication token (required for explicit authentication scheme)</li> </ul>
<b>Request</b>	The following items must be provided in <b>&lt;REQUEST-DATA&gt;</b> : <ul style="list-style-type: none"> <li>▪ <b>Mkey</b> – the merchant key associated with the account where transactions need to be settled</li> </ul>
<b>Response</b>	The action returns the following items in <b>&lt;RESPONSE-DATA&gt;</b> : <ul style="list-style-type: none"> <li>▪ <b>Status</b> – the status of the settlement operation</li> <li>▪ <b>Message</b> – additional text message that contain details about the settlement operation</li> </ul>