



US Equities Secure Web API

Version 1.1.0

April 5, 2011

Contents

1	Introduction	3
1.1	Overview.....	3
1.2	Requirements	3
1.3	Access	3
1.4	Obtaining a Developer Key and Secret	3
2	Request Structure	4
3	Response Structure	5
4	Services	6
4.1	Sponsored Access Service.....	6
4.1.1	viewClient	6
4.1.2	setClient	7
4.1.3	viewEasy	8
4.1.4	uploadEasy.....	8
4.1.5	removeEasy	8
4.1.6	viewRestricted	9
4.1.7	uploadRestricted	9
4.1.8	removeRestricted	9
5	Support	10

1 Introduction

1.1 Overview

The BATS Secure Web API allows client applications to view and update BATS data using the HTTPS protocol over the Internet. For example, if a client sponsors other clients, they can use the BATS Web Portal to manage these relationships. The API exposes this same functionality in a programmatic way allowing clients to write programs to automate these tasks. You could, for example, use the BATS API to limit the max share size allowed on all FIX sessions used by a particular sponsored client.

This API is available for use on both BATS BYX and BZX Exchange.

1.2 Requirements

Secure services:

- A BATS API Developer Key
- A BATS API Developer Secret
- Ability to send HTTPS requests and receive HTTPS responses over the Internet

1.3 Access

The API is accessed via a URL using the HTTPS protocol. The API is served from the api.batstrading.com domain.

BYX Exchange

<https://api.batstrading.com/byx/service/>

BZX Exchange

<https://api.batstrading.com/bzx/service/>

Where **service** is the path to a particular service you want to reach. The available services are detailed later in this document. Currently, only the *Sponsored Access* service is supported in the API. More services will be added later.

1.4 Obtaining a Developer Key and Secret

If you already have an API Key for the BATS Public Web API, you can use this same key. Contact the BATS Trade Desk to receive your API Key and Secret.

Never share your Secret with anyone. BATS associates will never ask you for your Secret.

2 Request Structure

Requests that retrieve data without making any modifications can be made using an HTTP GET. Requests that are expected to modify data must use an HTTP POST. Each service may have several commands available. Each command may have unique parameter requirements and so are specified per service and command. ***You need to submit your key and secret on every request.***

Example Request

For example, one of the services exposed via the API is the Sponsored Access service. One of the commands available in this service is “viewClient”. Since this command does not modify anything, you could use a GET.

BZX Service URL: https://api.batstrading.com/bzx/account/sponsored_access/

Method: GET

Required request parameters:

key: “dtiyt6VkQ1e3sFNgvuCaiA”

secret: “MyPassword”

command: “viewClient”

client: “ABCD”

Since this example is a GET, the API request could be made using this URL:

https://api.batstrading.com/bzx/account/sponsored_access/?key=dtiyt6VkQ1e3sFNgvuCaiA&secret=MyPassword&command=viewClient&client=ABCD

3 Response Structure

The response will be in JSON format. There are some key-value pairs that you can always expect while other keys will optionally exist. In the table below, the optional fields are shown in *italic*.

Additional key-value pairs and data items could be added. You should develop your application in such a way that it will not break if new items are added in the response. Do not rely on key position when parsing the response.

Field	Description
<code>code</code>	Request result code. <ul style="list-style-type: none">• 200 series = Success.• Non 200 series = Failure. See 'msg' value for details. The request may have been partially successful.
<code>msg</code>	A string description of the success or failure. Usually only populated when code is not 200.
<i>data</i>	A list of data items. The exact format is specific to the command requested.

Example Response Structure

```
{
  "code":"200",
  "msg": "",
  "data":[{"Color":"green","Size":12},{"Color":"blue","Size":9}]
}
```

4 Services

Currently, only the Sponsored Access service is supported in the Secure Web API.

Name	Path	Description
Sponsored Access	/account/sponsored_access/	Used by sponsors to monitor and limit sponsored client activity.

4.1 Sponsored Access Service

Service URL

BYX Exchange

https://api.batstrading.com/byx/account/sponsored_access/

BZX Exchange

https://api.batstrading.com/bzx/account/sponsored_access/

4.1.1 viewClient

Use this command to retrieve current settings for a specific sponsored client.

Parameter	Description
command	viewClient
client	BATS ID of the sponsored client

Example Response

```
{"code": "200", "data": [ {"allowShortSales": "0", "maxShareSize": 999999, "maxNotionalValue": 0, "allowPostmarket": "0", "allowIso": "1", "blockNewOrders": "0", "allowPremarket": "0"} ]}
```

4.1.2 setClient

Use this command to update settings for a specific sponsored client.

Parameter	Description
command	setClient
client	BATS ID of the sponsored client
<i>allowIso</i>	0=restrict, 1=allow
<i>allowPremarket</i>	0=restrict, 1=allow
<i>allowShortSales</i>	0=restrict, 1=allow
<i>blockNewOrders</i>	0=do not block, 1=block
<i>cancelOpenOrders</i>	0=do not cancel, 1=cancel
<i>maxNotionalValue</i>	0=use default max notional
<i>maxShareSize</i>	0=use default max size

**Optional parameters in italic. Note that at least one of the optional parameters must be specified. You may supply one or multiple optional parameters in a single request.*

Example Response

```
{"code": "200", "msg": ""}
```

4.1.3 viewEasy

Use this command to view the Easy to Borrow list for a specific sponsored client.

Parameter	Description
command	viewEasy
client	BATS ID of the sponsored client

Example Response

```
{"code":"200", "msg":"","data":{"known":["symbol1", "symbol2", "symbol3"], "unknown": [], "knownCount": 3, "unknownCount": 0}}
```

4.1.4 uploadEasy

Use this command to change the Easy to Borrow list for a specific sponsored client.

Parameter	Description
command	uploadEasy
client	BATS ID of the sponsored client
symbols	Comma separated list of BATS symbol names

Example Response

```
{"code":"200", "msg": "5 valid symbols were applied. Of these, 1 are not known to the BATS trading system. Perhaps they will be known in the future. ", "data": {"knownCount": 4, "unknownCount": 1, "unknown": ["FOOO"], "invalidCount": 0, "invalid": []}}
```

4.1.5 removeEasy

Use this command to remove the Easy to Borrow list for a specific sponsored client.

Parameter	Description
command	removeEasy
client	BATS ID of the sponsored client

Example Response

```
{"code":"200", "msg": ""}
```

4.1.6 viewRestricted

Use this command to view the Restricted Symbols list for a specific sponsored client.

Parameter	Description
command	viewRestricted
client	BATS ID of the sponsored client

Example Response

```
{"code": "200", "msg": "", "data": {"known": ["symbol1", "symbol2", "symbol3"], "unknown": [], "knownCount": 3, "unknownCount": 0}}
```

4.1.7 uploadRestricted

Use this command to change the Restricted Symbols list for a specific sponsored client.

Parameter	Description
command	uploadRestricted
client	BATS ID of the sponsored client
symbols	Comma separated list of BATS symbol names

Example Response

```
{"code": "200", "msg": "5 valid symbols were applied. Of these, 1 are not known to the BATS trading system. Perhaps they will be known in the future. ", "data": {"knownCount": 4, "unknownCount": 1, "unknown": ["FOOO"], "invalidCount": 0, "invalid": []}}
```

4.1.8 removeRestricted

Use this command to remove the Restricted Symbols list for a specific sponsored client.

Parameter	Description
command	removeRestricted
client	BATS ID of the sponsored client

Example Response

```
{"code": "200", "msg": ""}
```

5 Support

Please e-mail questions or comments regarding this specification to tradedesk@batstrading.com.

Revision History

Document Version	Date	Description
1.0.0	12/14/09	Release 1.0.0 distributed.
1.0.1	04/15/10	Noted support only for BZX Exchange at this time.
1.1.0	04/05/11	Added support for BYX Exchange.