**API**

**WHAT IS API?**

Bitstamp application programming interface (API) allows our clients to access and control their accounts, using custom written software.

**REQUEST LIMITS**

Do not make more than 600 request per 10 minutes or we will ban your IP address. For real time data please refer to the [websocket API](https://www.bitstamp.net/websocket/).

**PUBLIC DATA FUNCTIONS**

**TICKER**

*GET https://www.bitstamp.net/api/ticker/*

Returns JSON dictionary:

* last - last BTC price
* high - last 24 hours price high
* low - last 24 hours price low
* vwap - last 24 hours volume weighted average price: [vwap](http://en.wikipedia.org/wiki/Volume-weighted_average_price" \t "_blank)
* volume - last 24 hours volume
* bid - highest buy order
* ask - lowest sell order

**ORDER BOOK**

*GET https://www.bitstamp.net/api/order\_book/*

Returns JSON dictionary with "bids" and "asks". Each is a list of open orders and each order is represented as a list of price and amount.

**TRANSACTIONS**

*GET https://www.bitstamp.net/api/transactions/*

Params:

* time - time frame for transaction export ("minute" - 1 minute, "hour" - 1 hour). Default: hour.

Returns descending JSON list of transactions. Every transaction (dictionary) contains:

* date - unix timestamp date and time
* tid - transaction id
* price - BTC price
* amount - BTC amount

**EUR/USD CONVERSION RATE**

*GET https://www.bitstamp.net/api/eur\_usd/*

Returns JSON dictionary:

* buy - buy conversion rate
* sell - sell conversion rate

**API AUTHENTICATION**

All private API calls require authentication. You need to provide 3 parameters to authenticate a request:

* API key
* Nonce
* Signature

**API KEY**

To get an API key, go to "Account", "Security" and then "API Access". Set permissions and click "Generate key".

**NONCE**

Nonce is a regular integer number. It must be increasing with every request you make. Read more about it [here](http://en.wikipedia.org/wiki/Cryptographic_nonce). Example: if you set nonce to 1 in your first request, you must set it to at least 2 in your second request. You are not required to start with 1. A common practice is to use [unix time](http://en.wikipedia.org/wiki/Unix_time" \t "_blank) for that parameter.

**SIGNATURE**

Signature is a HMAC-SHA256 encoded message containing: nonce, client ID and API key. The HMAC-SHA256 code must be generated using a secret key that was generated with your API key. This code must be converted to it's hexadecimal representation (64 uppercase characters).

*Example (Python):  
message = nonce + client\_id + api\_key  
signature = hmac.new(API\_SECRET, msg=message, digestmod=hashlib.sha256).hexdigest().upper()*

**PRIVATE FUNCTIONS**

**ACCOUNT BALANCE**

*POST https://www.bitstamp.net/api/balance/*

Params:

* key - API key
* signature - signature
* nonce - nonce

Returns JSON dictionary:

* usd\_balance - USD balance
* btc\_balance - BTC balance
* usd\_reserved - USD reserved in open orders
* btc\_reserved - BTC reserved in open orders
* usd\_available- USD available for trading
* btc\_available - BTC available for trading
* fee - customer trading fee

**USER TRANSACTIONS**

*POST https://www.bitstamp.net/api/user\_transactions/*

Params:

* key - API key
* signature - signature
* nonce - nonce
* offset - skip that many transactions before beginning to return results. Default: 0.
* limit - limit result to that many transactions. Default: 100. Maximum: 1000.
* sort - sorting by date and time (asc - ascending; desc - descending). Default: desc.

Returns descending JSON list of transactions. Every transaction (dictionary) contains:

* datetime - date and time
* id - transaction id
* type - transaction type (0 - deposit; 1 - withdrawal; 2 - market trade)
* usd - USD amount
* btc - BTC amount
* fee - transaction fee
* order\_id - executed order id

**OPEN ORDERS**

*POST https://www.bitstamp.net/api/open\_orders/*

Params:

* key - API key
* signature - signature
* nonce - nonce

Returns JSON list of open orders. Each order is represented as dictionary:

* id - order id
* datetime - date and time
* type - buy or sell (0 - buy; 1 - sell)
* price - price
* amount - amount

**CANCEL ORDER**

*POST https://www.bitstamp.net/api/cancel\_order/*

Params:

* key - API key
* signature - signature
* nonce - nonce
* id - order ID

Returns 'true' if order has been found and canceled.

**BUY LIMIT ORDER**

*POST https://www.bitstamp.net/api/buy/*

Params:

* key - API key
* signature - signature
* nonce - nonce
* amount - amount
* price - price

Returns JSON dictionary representing order:

* id - order id
* datetime - date and time
* type - buy or sell (0 - buy; 1 - sell)
* price - price
* amount - amount

**SELL LIMIT ORDER**

*POST https://www.bitstamp.net/api/sell/*

Params:

* key - API key
* signature - signature
* nonce - nonce
* amount - amount
* price - price

Returns JSON dictionary representing order:

* id - order id
* datetime - date and time
* type - buy or sell (0 - buy; 1 - sell)
* price - price
* amount - amount

**WITHDRAWAL REQUESTS**

*POST https://www.bitstamp.net/api/withdrawal\_requests/*

Params:

* key - API key
* signature - signature
* nonce - nonce

Returns JSON list of withdrawal requests. Each request is represented as dictionary:

* id - order id
* datetime - date and time
* type - (0 - SEPA; 1 - bitcoin; 2 - WIRE transfer)
* amount - amount
* status - (0 - open; 1 - in process; 2 - finished; 3 - canceled; 4 - failed)
* data - additional withdrawal request data

**BITCOIN WITHDRAWAL**

*POST https://www.bitstamp.net/api/bitcoin\_withdrawal/*

Params:

* key - API key
* signature - signature
* nonce - nonce
* amount - bitcoin amount
* address - bitcoin address

Returns JSON dictionary if successful:

* id - withdrawal id

**BITCOIN DEPOSIT ADDRESS**

*POST https://www.bitstamp.net/api/bitcoin\_deposit\_address/*

Params:

* key - API key
* signature - signature
* nonce - nonce

Returns your bitcoin deposit address.

**UNCONFIRMED BITCOIN DEPOSITS**

*POST https://www.bitstamp.net/api/unconfirmed\_btc/*

Params:

* key - API key
* signature - signature
* nonce - nonce

Returns JSON list of unconfirmed bitcoin transactions. Each transaction is represented as dictionary:

* amount - bitcoin amount
* address - deposit address used
* confirmations - number of confirmations

**RIPPLE WITHDRAWAL**

*POST https://www.bitstamp.net/api/ripple\_withdrawal/*

Params:

* key - API key
* signature - signature
* nonce - nonce
* amount - currency amount
* address - bitcoin address
* currency - currency

Returns true if successful.

**RIPPLE DEPOSIT ADDRESS**

*POST https://www.bitstamp.net/api/ripple\_address/*

Params:

* key - API key
* signature - signature
* nonce - nonce

Returns your ripple deposit address.