Bulk REST API

The Authentication Token must be sent as the "x-authorization" HTTP Header.

This API uses a "Live" API key.

Please see Accepted CSV Fields for a description of the columns that are accepted in the CSV file being uploaded.

The more data fields included, the better the records can be analyzed.

Important Information:

If the "export_type" is "single" the results are not 'segmented' in the same manner "multi". It is up to the implementor to examine the results.csv and handle the data accordingly (including separating out any blacklisted entries).

We recommend only contacting results with the following properties:

1. subscriber_score_bucket of 'high'
2. subscriber_email_status of 'verified'
3. No blacklist hits (examine the following fields: subscriber_email_is_blacklisted, subscriber_email_domain_is_blacklisted, subscriber_signup_domain_is_blacklisted, subscriber_signup_ip_is_blacklisted)

File Format:

Any file being uploaded must be a correctly formatted Comma Separated Value file (CSV) using *UTF-8 encoding*. See RFC 4180. At a minimum, Email is required. We suggest including CAN-SPAM required fields and the API Key can be configured to enforce those requirements. For a list of allowed fields and their descriptions see Accepted CSV Fields.

We have a desktop application available for formatting CSV files available (contact your account manager).

1. The file *must* have a valid header row to indicate which fields are in which columns. See the table of Accepted Fields for a list of acceptable values.

2. Columns must be separated by commas ","

3. All rows must be terminated using Windows line endings (CRLF \r\n)

4. All rows must have same number of columns

5. Values containing commas must be quoted with double-quotes (")

   e.g.

   Before: 123 test way, apt #4

   After: "123 test way, apt #4"

6. Values containing double quotes must be quoted with double-quotes (") and the double-quotes within the value must be escaped with a double-quote

   e.g.

   Before: test"one

   After: "test""one"

7. Do not send the string "NULL" if you do not have a value for a field. Just send an empty value.

8. Be careful with time stamps and dates. If you're storing your time stamps and dates as a unix time integer, make sure to *not* send 1970-01-01 00:00:00. Just send an empty value

Endpoints:

IMPORTANT The API Endpoint when viewing the Auth details of the API Key will end in 'verify'. This *must* be changed to 'bulk'

GET to /api/live/bulk (List jobs)

POST to /api/live/bulk (Create Job)

GET to /api/live/bulk/ID (Job Status)

DELETE to /api/live/bulk/ID (Delete job files)

GET to /api/live/bulk/ID/download (Download Job Results)

Errors:

Any error, no matter the HTTP code, should return a JSON object in the form:

```
{
    "status": "error",
    "error" : "CODE",
    "msg" : "HUMAN_READABLE_MESSAGE"
}
```

List all Jobs [200]:

No Jobs:

```
{
    "data":[],
    "total":0
}
```

With Jobs:

```
{
    "data":[
        {
            "download_url":"http://host:port/api/live/bulk/ID/download",
            "job":"ID1",
            "added":"2014-09-19 17:17:46 UTC",
            "started":"2014-09-19 17:18:30 UTC",
            "ended":"2014-09-19 17:19:05 UTC",
            "status":"completed"
        },
        {
            "job":"ID2",
```

```
        "added":"2014-09-19 17:17:43 UTC",
        "started":"2014-09-19 17:17:44 UTC",
        "status":"active"
    },
    {
        "job":"ID3",
        "added":"2014-09-19 17:17:44 UTC",
        "status":"pending"
    }
    ],
    "total":3
}
```

Possible job statuses:

- active
- cancelled
- completed
- paused
- pending

Create a new job:

Creating a job is done as a multipart form POST (encoding multipart/form-data).

Parts of the POST operation:

1. **export_type:** this can be "multi" or "single" (defaults to "multi"). This setting determines how the results are segmented, with "multi" outputing 'high', 'medium', 'low', 'bad' and 'blacklisted' CSVs, and "single" outputting just 'results.csv'.
2. **notify_email:** An email address which will be notified when the job is completed (optional)
3. **max_verify_hours:** The number of hours after which the verification process will be stopped (optional). **When this happens, unverified records will be included in the bad.csv file with a 'notverified' email_status.**
4. the file, which must be of type "text/csv" (required)
5. The **"x-authorization" HTTP header** must be set for all operations. The value of this field is the API Key Authentication Token

Example form

In all likelyhood you will not have a real form, but will be submitting the file programmatically. This example is just to indicate how the content of the POST might be constructed. It does *not* set the correct 'x-authorization' HTTP header value.

```html
<form action="http://api.example.com:8080/api/live/bulk" method="POST" enctype="multipart/form-data">
  <select name="export_type" id="export_type">
    <option value="multi">Export mutiple files segmented by score</option>
    <option value="single">Export a single 'results.csv' file</option>
  </select>
  <!-- Send an email to 'test@example.com' when the job is complete -->
  <input type="text" name="notify_email" id="notify_email" value="test@example.com" />
  <!-- Stop processing if the job runs for more than 6 hours -->
```

```html
<input type="text" name="max_verify_hours" id="max_verify_hours" value="6" />
<input type="file" name="file" id="file" />
</form>
```

Responses

Success Response [202] (when file is < 500k lines):

```json
{
  "status":"success",
  "jobs":[
    {
      "name":"ID",
      "msg":"FILENAME being processed as a single jobs",
      "status_url":"http://host:port/api/live/bulk/ID"
    }
  ]
}
```

Success Response [202] (when file is > 500k lines)
    The system will split the file into multiple jobs)
    When a file is > 500k lines, the system will automatically split it into
    multiple jobs and append a counter to the job name.
    The counter will take the form of '_00', incrementing from 00 to 99.

```json
{
  "status":"success",
  "jobs":[
    {
      "name":"ID",
      "msg":"FILENAME being processed as a multiple jobs",
      "status_url":"http://host:port/api/live/bulk/ID"
    },
    {
      "name":"ID2",
      "msg":"FILENAME being processed as a multiple jobs",
      "status_url":"http://host:port/api/live/bulk/ID2"
    }
  ]
}
```

Failure Response [400]:

```json
{
  "status":"error",
  "error":"bad_parameter",
  "msg":"Missing CSV file in POST"
}
```

An error should always contain 'status' as 'error', the 'error' code and a readable message in 'msg'

Possible error codes:

- not_found
- bad_parameter
- already_deleted
- missing
- out_of_credits

Get the status of a Job:

Failure [404] if the job is not found:

```
{
  "status" : "error",
  "error" " "not_found",
  "msg" : "Cannot find job ID"
}
```

Success [200]:

```
{
  "job":"ID",
  "added":"2014-09-19 17:49:20 UTC",
  "started":"2014-09-19 17:49:21 UTC",
  "status":"active"
}
```

and when it's completed

```
{
  "download_url":"http://host:port/api/live/bulk/ID/download",
  "job":"ID",
  "added":"2014-09-19 17:49:20 UTC",
  "started":"2014-09-19 17:49:21 UTC",
  "ended":"2014-09-19 17:49:57 UTC",
  "status":"completed"
}
```

Delete the source and results files for a job once you've downloaded successfully:

Failure [404] if the job is not found

```
{
  "status" : "error",
  "error" " "not_found",
  "msg" : "Cannot find job ID"
}
```

Failure [400] if the job cannot be deleted

```
{
    "status" : "error",
    "error" : "Not_completed",
    "msg" : "Only completed jobs can have files deleted"
}
```

Success [200]

```
{
    "status" : "deleted",
    "msg" : "All files for job \"ID\" have been deleted"
}
```

Download the results when the job is completed:

Failure [404] - Job not found

```
{
    "status" : "error",
    "error" " "not_found",
    "msg" : "Cannot find job results file for ID"
}
```

Failure [404] - Job already deleted

```
{
    "status" : "error",
    "error" " "already_deleted",
    "msg" : "Files for ID already deleted"
}
```

Failure [404] - Not ready for download

```
{
    "status" : "error",
    "error" " "not_ready",
    "msg" : "File is not available for download until the job has completed"
}
```

Failure [409] - Requesting wrong data

```
{
    "status" : "error",
    "error" " "conflict",
    "msg" : "File not available for this job type"
}
```

Download the results when the job is not yet completed:

Returns status [200]

```
{
  "status" : "X",
  "msg" : "Job \"ID\" is currently \"X\""
}
```

Where X is 'pending', 'paused', 'active', 'completed', 'cancelled'

Accepted CSV Fields:

The CSV must contain a header row which consists of field names matching those listed below.

Field	Description
subscriber_email	Email Address **required**
subscriber_fname	First Name **suggested**
subscriber_mname	Middle Name or Initial(s)
subscriber_lname	Last Name **suggested**
subscriber_addr1	Address Line 1
subscriber_addr2	Address Line 2
subscriber_phone	Phone Number
subscriber_mobile	Mobile Phone Number
subscriber_fax	Fax Number
subscriber_city	City **suggested**
subscriber_state	State/Region Code **suggested**
subscriber_zip	Zip/Postal Code **suggested**
subscriber_country	Country Code (2 character) **suggested**
subscriber_gender	Gender (M/F) **suggested**
subscriber_dob	Date of Birth. "YYYY-MM-DD" format **suggested**
subscriber_signup_ip	Sign up IP Address **suggested**
subscriber_signup_tstamp	Sign up Time stamp "YYYY-MM-DD hh:ii:ssZ" Format **suggested**
subscriber_signup_url	Sign up Web page URL **suggested**
subscriber_signup_category	Sign up Web page Category (See Categories below) **suggested**
subscriber_last_activity_tstamp	Time stamp of when subscriber last interacted. "YYYY-MM-DD hh:ii:ssZ" Format
subscriber_reference	Used for matching records between systems
custom_	Any header starting with "custom_" *

* Custom Headers: will be passed through from the input file into the results file in the correct position

Categories

Automotive

      Automotive::Maintenance

    Automotive::Parts

    Automotive::Sales

Bizop

    Bizop::Make Money From Home

Business Services

Careers & Employment

Computers & Communications

    Computers & Communications::Cable Bundle

    Computers & Communications::Cable TV

    Computers & Communications::Cellular/Mobile Packages

    Computers & Communications::Cellular/Mobile Phones

    Computers & Communications::Computer Hardware

    Computers & Communications::Computer Software

    Computers & Communications::Consulting Services

    Computers & Communications::Internet Services

    Computers & Communications::Satellite TV

    Computers & Communications::Video Games

    Computers & Communications::VOIP

Consumer Packaged Goods

    Consumer Packaged Goods::Food products

    Consumer Packaged Goods::Household products

    Consumer Packaged Goods::Packaged goods

    Consumer Packaged Goods::Tobacco

Dating

    Dating::Adult Dating

    Dating::Christian Dating

    Dating::International Dating

    Dating::Senior Dating

Education

    Education::Accounting Program

    Education::Art & Design Program

Education::Business Program

Education::Education Program

Education::Health Science Program

Education::Health Science Program::Dental Assistant Degree

Education::Health Science Program::Medical Admin Degree

Education::Health Science Program::Nursing Degree

Education::Health Science Program::Pharmacy Assistant Degree

Education::Health Science Program::Ultrasound Tech Degree

Education::Health Science Program::X-ray Tech Degree

Education::High School Diploma

Education::Legal Program

Education::Legal Program::Criminal Justice Degree

Education::Legal Program::Paralegal Degree

Education::Online Degree

Education::Technology Program

Entertainment

Entertainment::Amusement

Entertainment::Box Office

Entertainment::Film

Entertainment::Music

Entertainment::Recreation

Events

Family

Financial

Financial::Auto Loans

Financial::Business credit institutions

Financial::Commercial banks

Financial::Consumer finance companies

Financial::Credit agencies

Financial::Credit Card

Financial::Credit card agencies

Financial::Credit Repair

Financial::Debt Consolidation

Financial::Grants

Financial::Grants::Education Grant

Financial::Grants::Government Grant

Financial::Loan companies

Financial::Payday Loans

Financial::Personal credit

Financial::Personal Loan

Financial::Refinance

Financial::Securities

Financial::Student Loan

Financial::Taxes

Financial::Underwriting

Financial::VA Loans

Fundraisers

Health & Beauty

Health & Beauty::Anti-aging

Health & Beauty::Cosmetics

Health & Beauty::Diet & Weight Loss

Health & Beauty::Fitness

Health & Beauty::Medical Services

Health & Beauty::Pharmaceuticals

Health & Beauty::Skin Care

Health & Beauty::Toiletries

Insurance

Insurance::Auto Insurance

Insurance::Dental Insurance

Insurance::Health Insurance

Insurance::Home Insurance

Insurance::Life Insurance

Insurance::travel Insurance

Legal Services

Leisure & travel

Leisure & travel::Airline

Leisure & travel::Hotel

Leisure & travel::Resort

Leisure & travel::travel

Media

Media::Books

Media::Magazines

Media::Newspapers

Media::Radio

Media::TV

Newsletters

Pets

Pharma & Healthcare

Promotional Offer

Real Estate

Retail

Retail::Apparel

Retail::Appliances

Retail::Coupons

Retail::Drugstores

Retail::Fast food

Retail::Home furnishings

Retail::Home & Garden

Retail::Jewelry

Retail::Mail order/catalog

Retail::Pet food/supplies

Retail::Restaurants

Retail::Retail stores

Retail::Textiles

Retail::Toys

Telecommunications

Warranty

Warranty::Auto Warranty

Warranty::Home Warranty