

Peter Gustafson writing sample
TowerData
December 19,2016

title: TowerData Bulk Email Validation API Reference

toc_footers:

- Contact TowerData

includes:

search: true

Bulk API Overview

TowerData's Bulk Email Validation API uses the industry standard REST syntax, employing asynchronous API design pattern to facilitate lengthy jobs. Data is returned in JSON format, which is both easy for you to read and easy for your programs to analyze.

The API accepts bulk email list data in JSON as well as CSV/TSV tabular dataset where a column of email addresses exists.

For single email validation API, please refer to [Email Intelligence and Validation API](http://docs.towerdata.com/)

When to Use Bulk API

Bulk API is based on REST and async API design principles and is optimized for processing large sets of data.

You can use it to submit many email addresses by submitting them in one batch.

While the API service processes batches in background, you can check progress by using a job specific URL provided in the immediate response to track progress and get metrics as the progress goes on.

Please note that the recommended size of entries should be less than 1M;

although the API can handle larger payloads, it will also take longer time to get all the results, especially for email validation.

<aside class="notice">

All developers need agree on the following Terms of Service.

</aside>

[Terms of Service](http://intelligence.towerdata.com/developers/data-services-terms-and-conditions)

API Endpoint

> ****API Endpoint****

>

> <https://api.towerdata.com/v6/bulk>

The original file submission URL:

* <https://api.towerdata.com/v6/bulk>

With HTTP POST request to submit user data and with a valid api key.

Then the job status API Endpoint is located at:

* https://api.towerdata.com/v6/bulk/{list_id}

With HTTP GET request to see the jobs and the status.

Authentication

> To authorize, you need a valid API Key:

> *The API Key is configured to perform certain service, and the data you'll receive is decided by the API Key.*

To register on <https://dashboard.towerdata.com/> to get an API key, where you can get a default set of fields and then please [contact us](info@towerdata.com) to add any additional fields.

TowerData expects for the API key to be included in all API requests to the server in the parameter:

For bulk API, it should be in the headers and it looks like:

```
`TD-API-Key:{your-api-key}`
```

Bulk API Request & Response

Asynchronous Request

You first submit the file via a request to get a handle URL to check further information, such as job status.

Payload (the data) of the request should be in the post body:

1. For payload in In JSON, set Content-Type header to "application/json"
2. For payload in File input, set Content-Type header to "Content-Type: text/csv; charset=UTF-8 (gray means optional, default)"

```
<aside class="success">
```

```
Example:
```

```
curl -X POST --header "TD-API-Key:78ad9ddc21e3c220cc5da024b6dbe13c;
Content-Type:text/csv"
https://api.towerdata.com/v5/bulk?
type=validation&email_column=1&start_at=1 -d @myfile.txt
</aside>
```

```
## Query Parameters
```

```
Parameter|Description|Example
```

```
-----|-----|-----
```

```
type|Request type: validation, intelligence, appending. Default:
```

```
validation|type=validation
```

```
timeout|The max time allowed to check single item (email) in seconds,
default=30|timeout=10
```

```
email_column|If the input is a tabular dataset (CSV/TSV etc.), the
index of column of email addresses, it is 1-based. Default=1|
```

```
email_column=1
```

```
start_at|Which row to start processing, 1-based. If exists the header
row, set start_at=1. Default=1|start_at=1
```

```
debug|To return additional message. TBD Default=false|debug=true
```

```
delimiter|For file, default is TAB (\t)|delimiter=","
```

```
save|If we save the payload to a file. Default save=true. V1 save
always true|save=true
```

```
name|A custom name for customer to use for their input file,
default="myfile"; we will keep the input as the file for up to N days
(N per license agreement). If filename=null, we will use the default.|
name="abc"
```

```
## Original Response
```

```
If the request was successful, you will get:
```

```
```rest
```

```
202 Accepted
```

```
```
```

```
with Header
```

```
```rest
```

```
Location: https://api.towerdata.com/v5/bulk/{list_id}
```

```
```
```

```
The *list_id* is the identifier of the list of email addresses you
submitted via the previous request.
```

```
You also can find the information in the body of response. It looks
like:
```

```
```json
```

```
{
```

```
 "list_id": 53424, // the resource id (ids_file_id)
```

```
 "jobs": ["81545v12353252542"], // the jobs array
```

```
 "payload": "JSON", // payload type JSON | FILE
```

```
 "size": 654321 // size of the file/payload
```

```
}
...
```

## ## Job Status Request

Now you can retrieve the jobs directly from the original URL response (we return all jobs in one call).

You can see the jobs by calling HTTP GET on:

```
https://api.towerdata.com/v5/bulk/{list_id}/job?status={status}
&type={type}&expand=true
```

There are 3 filters: *\*status\**, *\*type\** and *\*expand\**.

Parameter	Description	Default Value
status	the job status	{all jobs}
type	the type of jobs	{all types}
expand	including job details	false

If no status parameter is present, return all jobs regardless of status;  
if no type is set, return all types of jobs;  
and expand=true will retrieve a list of job with details, otherwise, only the job ids.

### ### Example

For example, the response of

```
* https://api.towerdata.com/v5/bulk/{list_id}/job?expand=true
```

may look like:

```
```rest  
HTTP/1.1 200 OK  
HEADER: Last-Modified=xxx; Date=xxx; ...  
BODY:  
{  
  [  
    {"status":PENDING,  
     "job_id":"123456v1324565979","input_count"=34521,"completed_count"=0},  
    {"status":IN-PROGRESS,  
     "job_id":"123456v1324565989","input_count"=34521,"completed_count"=235  
41},  
    {"status":DONE,  
     "job_id":"123456v1324565969","input_count"=34521,"completed_count"=235  
41}  
  ]  
}  
```
```

If the *\*expand\** parameter is missing (the default value is false), the

response to

\* [https://api.towerdata.com/v5/bulk/{list\\_id}/job?status=IN-PROGRESS](https://api.towerdata.com/v5/bulk/{list_id}/job?status=IN-PROGRESS)

will look like:

```
```json
{"jobs": [
  "123456v1484565978",
  "123456v1486565969"
]}
```
```

## Job Status Request and Response

You can further check each single job's status by:

\* GET: [https://api.towerdata.com/v5/bulk/{list\\_id}/job/{job\\_id}](https://api.towerdata.com/v5/bulk/{list_id}/job/{job_id})

and you should add api-key in TD-Api-Key header in requesting job status as usual.

If successful, the job status URL will respond:

For calls to <https://api.towerdata.com/v5/bulk/15432/job/123456v1324565969> with correct API key in the header, if successfully, you will get:

\* When status=PENDING (0)

```
```rest
HTTP/1.1 200 OK
HEADER: Last-Modified=xxx; Date=xxx; ...
BODY: {"status":PENDING, "job_id":"123456v1324565969"}
```

When status=IN-PROGRESS(1)

```
HTTP/1.1 200 OK
HEADER: Last-Modified=xxx; Date=xxx; ...
BODY:
{
  "status": IN-PROGRESS,
  "job_id": "123456v1324565969",
  "input_count": 99838,
  "completed_count": 57364,
  "total_time_elapsed": 222546,
  "min_record_time": 7,
  "max_record_time": 5000,
  "average_record_time": 89,
}
```

* When status=DONE(2)

```

```rest
HTTP/1.1 303 See Other
HEADER:
Last-Modified=xxx;
Date=xxx;
Location=https://api.towerdata.com/v5/bulk/15432/result
...
BODY: the same as before except status=DONE
And
BODY:
{
 "status": DONE,
 "job_id": "123456v1324565969",
 "input_count": 99838,
 "completed_count": 57364,
 "total_time_elapsed": 222546,
 "min_record_time": 7,
 "max_record_time": 5000,
 "average_record_time": 89,
 "metrics": {
 "overall": {
 "valid": 34568,
 "invalid": 3456,
 "unknown": 1834,
 "unverifiable": 897,
 "risky": 352,
 "corrected": 132
 }
 }
 "invalid_breakdown": {
 "syntax_errors": 985,
 "domain_errors": 734,
 "mailbox_errors": 1235,
 "suppressed": 342
 }
 "role_account": 134,
 "domain_type": {
 "disposable": 32451,
 "government": 11234,
 "wireless": 2344,
 "freeisp": 3434
 }
}
}
...

```

Please note that in this case, in the header there is a location property:

Location=<https://api.towerdata.com/v5/bulk/15432/result>

it is where you need to retrieve the final result. Please refer to [Final Result](#Final\_Result) section for more info.

\* When status=FAILED(-1) / ABORTED(-2) / ERROR(-3)

```
```rest
HTTP/1.1 200 OK
HEADER: Last-Modified=xxx; Date=xxx; ...
BODY:
{
  "status":FAILED/ABORTED/ERROR,
  "Job_id":"123456v1324565969",
  "Status_code": 1001,
  "detailed_message":"Invalid input file."
}
```
```

### ### Job Status

The following is a list of job statuses:

- \* PENDING: the job has not started yet, possibly wait for previous job in the pipeline to finish
- \* IN-PROGRESS: the job is started and is in progress. You should be able to see the progress by the status URL
- \* DONE: the job is finished successfully. The result should be ready to download and the metrics available too.
- \* FAILED: the job is failed, and error message should be given
- \* ABORTED: the job is canceled or stopped - either by user or by Towerdata. Reasons should be given.
- \* ERROR: the job encountered an unrecoverable error and has to stop. Reasons should be given.

### ### Job Types

The following is a list of job types

- \* email-validate: for email validation
- \* ei\_match: for email (and postal) intelligence
- \* importlist: importing user file/email list
- \* countlist: counting and preprocessing the user file/list
- \* parselist: parsing the user file/list

### # Bulk API Result

Once you have the URL to the final result, and the job is finished successfully. You can use the URL (GET) to retrieve result.

The result URL should look like this, requested with proper headers with api-key included:

GET: <https://api.towerdata.com/v5/bulk/15432/result/>

Whether original request is FILE or JSON payload, you will get a response with status 200 OK with:

- \* the headers as Last-Modified={updated\_at} and Date={now()}
- \* Content Type as application/csv
- \* and a header 'Download-Link={secure-link-to-use-in-browser}'.

<!--

See below for more detailed example in case of JSON.

In the case of success, the response can be:

```
```json
[
  {
    "address":"jwang@towerdata.com",
    "corrections":[
      "jwang@towerdata.com"
    ],
    "ok":false,
    "validation_level":2,
    "status":"invalid",
    "status_code":155,
    "domain_type":null,
    "role":false
  },
  {
    "address":"jwang@gmail.com",
    "ok":false,
    "validation_level":5,
    "status":"invalid",
    "status_code":330,
    "domain_type":"freeisp"
  },
  {
    "address":"unknown@example.com",
    "ok":true,
    "validation_level":2,
    "status":"unverifiable",
    "status_code":45,
    "domain_type":"business",
    "role":false
  },
  {
    "address":"NOT AN EMAIL ADDRESS",
    "ok":false,
    "validation_level":2,
    "status":"invalid",
    "status_code":130,
    "domain_type":null,
    "role":false
  }
]
```

```
]
...
-->
```

The header contains:

```
Last-Modified:{date time}
Content-Type: application/json
...
```

FILE download case

To facilitate browser users, we also provide a link to be used in browser windows.

The Download-Link is for this purpose.

The file download URL looks like:

```
https://api.towerdata.com/v5/bulk/15324/result/download?
token=30cx02305089238570x303598543:7ueyxb83983u837d
```

and you will see the file starts downloading.

Bulk API Code Examples

cURL

> cURL

```cURL

```
curl -X POST --header "TD-API-Key:78ad9ddc21e3c220cc5da024b6dbe13c;
Content-Type:text/csv"
```

```
https://api.towerdata.com/v5/bulk/csv?email_column=1&start_at=1 -d
@myfile.txt
```

```

Sample code calling EV API with curl.

Python

> python

```python

```
import requests
```

```
url = "https://api.towerdata.com/v5/td"
```

```
querystring = {"email_column":1,"start_at":1}
```

```
headers = {'TD-API-Key':'78ad9ddc21e3c220cc5da024b6dbe13c', 'Content-
Type':'text/csv'}
```

```
file = open('myfile.txt', 'r')
```

```
payload = file.readlines()
```

```
response = requests.request("POST", url, params=querystring,
headers=headers,payload=payload)
check response code if necessary
print(response.text)
```
```

Sample code calling EV API with Python.

```
## Java
```

```
> java
```

```
```java
HttpResponse<String> response = Unirest.post("http://
api.towerdata.com/v6/bulk")
 .header("content-type", "multipart/form-data; boundary=-----
WebKitFormBoundary7MA4YWxkTrZu0gW")
 .header("td-api-key", "deadbeefdeadbeef1234567890")
 .header("email_column", "1")
 .header("type", "validation")
 .header("start_at", "1")
 .header("cache-control", "no-cache")
 .body(file_content)
 .asString();
```
```

Sample code calling EV API with Java.

```
## JavaScript
```

```
> javascript
```

```
```javascript
var form = new FormData();
form.append("file", "");

var settings = {
 "async": true,
 "crossDomain": true,
 "url": "http://api.towerdata.com/v6/bulk",
 "method": "POST",
 "headers": {
 "td-api-key": "deadbeefdeadbeef1234567890",
 "email_column": "1",
 "type": "validation",
 "start_at": "1",
 "cache-control": "no-cache",
 },
 "processData": false,

```

```
 "contentType": false,
 "mimeType": "multipart/form-data",
 "data": form
}

$.ajax(settings).done(function (response) {
 console.log(response);
});
````
```

Sample code calling EV API with JavaScript.

```
## C`#`  
  
> C#  
  
`` `c#  
var client = new RestClient("http://api.towerdata.com/v6/bulk");  
var request = new RestRequest(Method.POST);  
request.AddHeader("cache-control", "no-cache");  
request.AddHeader("start_at", "1");  
request.AddHeader("type", "validation");  
request.AddHeader("email_column", "1");  
request.AddHeader("td-api-key", "deadbeefdeadbeef1234567890");  
IRestResponse response = client.Execute(request);  
````
```

Sample code calling EV API with C#.