



TraxCloud REST API Documentation

Prepared by:

Yann Salaün, Software Engineer

Jean-Denis Muys, R&D Director

October 6, 2016

SUMMARY

This document describes the TraxCloud REST API. Both the API and the documentation are in early stage. We are aware of some of the issues with both, and we fully intend to fix them. However, we may not be aware of all the issues, and the reader is encouraged to provide feedback to Audionamix.

1. Structure of the API

The API is designed as a REST API. It can only be accessed over the HTTPS protocol.

Protocol	HTTPS
Base URL	https://trax.audionamix.com/api/v1/

The API manages resources that can be manipulated in a CRUD way with URIs. To access a given *resource*, concatenate the resource name to the base URL, and append any optional variation, following the model:

https://trax.audionamix.com/api/v1/resource/schema/	the resource JSON schema
https://trax.audionamix.com/api/v1/resource/	the list of existing <i>resources</i>
https://trax.audionamix.com/api/v1/resource/id/	one existing <i>resource</i> with the given <i>id</i>

To run TraxCloud's automatic voice extraction, you will only need the following resources: *audiofile*, *configuration*, *preanalysis* and *separation*.

All the resources, and more generally, all the API responses, are JSON objects, encoded in UTF-8 with no BOM.

In this document, all strings set in *orange* are placeholders that must be replaced by actual values.

Note: audio files are represented in the API by the *audiofile* resource. At this point in time, only the .wav audio file format is supported by the API. The audio file must contain 16 bit samples, and the sample rate must not be higher than 44100 kHz. Lower sample rates are supported (and even recommended) with no degradation in the quality of the separation.

2. Authentication

The API authenticates clients, and gives clients access according to their access rights, using a single HTTP header, which provides the client's **user name**, and the client's **API key**. Your user name and API key should have been provided to you upon delivery of Audionamix software. If not, please contact Audionamix technical support at <mailto:support@audionamix.com>. The API key is a string composed of, as of this writing, 40 lower-case hexadecimal figures.

Header name	Authorization
Header value	ApiKey <username>:<api_key>
Example	Authorization: ApiKey username:api_key

Note that the space after “**ApiKey**” must be provided.

How you set the header depends on the technology you use to call the API. The examples below use the `curl` command line tool. With `curl`, headers are set using the `-H` option, for example:

```
curl -X GET -H "Authorization: ApiKey username:api_key" "https://trax.audionamix.com/api/v1/audiofile/"
```

For brevity, this authentication header will be omitted in all examples that follow.

3. Synopsis

This section shows the succession of calls for two cases: a simple separation, and a separation with a pre-analysis step.

3.1. Simple separation

3.1.1. Upload an audio file

To upload an audio file, you need to do a **POST** request to the **audiofile** resource with the content of your audio file set to the **file** field of a multipart body. The response's JSON **id** attribute is the identifier of the audio file and will be used during next step.

Example with `curl`:

```
curl -X POST "https://trax.audionamix.com/api/v1/audiofile/" -F file=@path/to/file.wav
```

3.1.2.Start separation

To start a separation, you need to do a **GET** request to the **separation** resource with **file_id** URL parameter set to the audio file identifier that you received above. This will start a separation as a background job. The response's JSON **id** attribute is the **separation** identifier and will be used during next step.

Example with curl:

```
curl -X GET "https://trax.audionamix.com/api/v1/separation/?file_id=audiofile_id"
```

3.1.3.Check separation status

You can check the progress of the separation job with a **GET** request on the **separation** resource you just created and look at the value of the **status** attribute in the JSON response. It shall be an integer value between 0 and 100. A value of 100 means that the separation is done and that you may download the resulting audio file, whose identifier is the **extracted_file_id** attribute.

Example with curl:

```
curl -X GET https://trax.audionamix.com/api/v1/separation/separation_id/
```

3.1.4.Download extracted audio file

To download the resulting audio file with the extracted sound, do a **GET** request to the **audiofile** resource with the **pk** URL parameter set to the audiofile id. This will redirect to the location of the audio file.

Example with curl (we use -L to follow the redirection and -o to name the local output file):

```
curl -X GET "https://trax.audionamix.com/api/v1/audiofile/?pk=extracted_audiofile_id" -L  
-o path/to/result.wav
```

3.2.Enhanced separation with pre-analysis

3.2.1.Upload an audio file

First you need to upload an audio exactly as above. You only have to upload the audio file once. You can use it several times to execute different separations.

3.2.2.Start pre-analysis

To start a pre-analysis, you need to do a **GET** request to the **preanalysis/start** subresource. Two URL parameters are required and one parameter is optional:

Parameter	Type	Default	Description
file_id	int	None	The input file id
algo	string	pitch	The algorithm that will run. Two choices are possible: "pitch" for pitch detection and 'csnt' for consonant detection
baseline	string	v1	The version of the algorithm. Optional parameter only used for the pitch detection algorithm. Two choices are possible: "v1" and "v3". "v3" is recommended for the best detection.

This will start a pre-analysis as a background job. The response's JSON **id** attribute is the identifier of the preanalysis and will be used during next step.

Example with curl (pitch algorithm):

```
curl -X GET "https://trax.audionamix.com/api/v1/preanalysis/start/?
file_id=audiofile_id&algo=pitch&baseline=v3"
```

Example with curl (consonant algorithm):

```
curl -X GET "https://trax.audionamix.com/api/v1/preanalysis/start/?
file_id=audiofile_id&algo=csnt"
```

3.2.3. Check pre-analysis status

Do a **GET** request to the **preanalysis** resource you just created and look at the value of the **status** attribute of the JSON response. It shall be an integer value between 0 and 100. A value of 100 means that the pre-analysis is done and that you may download the resulting annotation file.

Example with curl:

```
curl -X GET "https://trax.audionamix.com/api/v1/preanalysis/preanalysis_id/"
```

3.2.4. Download preanalysis result

To download the preanalysis result, you need to do a **GET** request to the **result** subresource of the **preanalysis** you created.

The result is a JSON object that has the following fields.

Field	Type	Description
big_dipper	bool	Whether the pitch detection algorithm has already run. This is required to make sure the separation algorithm don't run its own pitch detection step.
pitch_annotation	string	An array of pitch annotation segments. Each segment is an array of time-frequency points. Each point is an array of two floats: first is a time (in seconds) and second is a frequency (in Hz).
consonants_activity	string	An array of time points (in seconds). Each points goes by pair, the first being the start of a consonant segment and the second being the end of a consonant segment.

Example with curl (we use **-L** to follow the redirection and **-o** to name the local output file):

```
curl -X GET "https://trax.audionamix.com/api/v1/preanalysis/preanalysis_id/result/" -L -
o path/to/config.json
```

3.2.5. Create the configuration

The typical workflow is to run two preanalysis, with the pitch detection algorithm and the consonant detection algorithm, and then merge the two results. The resulting data structure is then used to inform the separation algorithm with pitch and consonant annotations. In addition, it is possible to add the following field to the configuration:

Field	Type	Default	Description
reverb_time	float	0.0	In seconds. To set in the range [0.5,3.0], or 0 for no reverb. Allows for the extraction of the reverberation together with the dry voice. Typical value is 1.0. Increases the duration of the computation up to a factor 10 when compared to a Standard Quality separation without reverberation

3.2.6. Upload the configuration

You need to upload the configuration to TraxCloud, using a **POST** request to the **configuration** resource with a JSON body being the configuration created at the previous step . The response's JSON **id** attribute is the identifier of the **configuration** and will be used during next step.

Example with curl:

```
curl -X POST "https://trax.audionamix.com/api/v1/configuration/" -H "Content-Type: application/json" -d @path/to/config.json
```

3.2.7.Start configured separation

Start a separation as shown in the "Simple separation" example, but with an additional URL form parameter, named **config_id**, containing the identifier of the configuration from the previous step.

Example with curl:

```
curl -X GET "https://trax.audionamix.com/api/v1/separation/?file_id=audiofile_id&config_id=configuration_id"
```

3.2.8.Check separation status

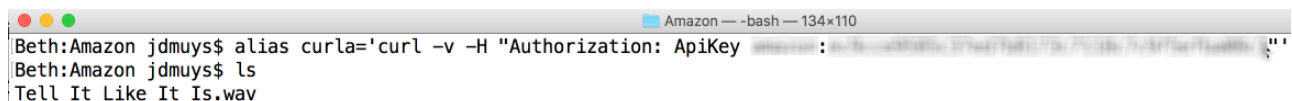
As above

3.2.9.Download extracted audio file

As above

4. Sample terminal session

This terminal session was run in a Mac OS X 10.11 terminal running the Bash shell. It uses the curl terminal command as the REST client. In order to avoid always typing the authentication header, we first define an alias:



```
Amazon -- bash -- 134x110
Beth:Amazon jdmuys$ alias curla='curl -v -H "Authorization: ApiKey [redacted]"'
Beth:Amazon jdmuys$ ls
Tell It Like It Is.wav
```

The **-v** parameter is optional. It makes curl "verbose". With this parameter, curl will show the HTTP requests as well as the responses.

With this alias defined, we only need to type **curla** instead of **curl**.

As you can see in the screen shot above, we have an audio file named "Tell It Like It Is.wav" in the current directory. We are going to run separations on this file. The file name contains spaces, which is slightly annoying in the terminal, but is otherwise not a problem.

In the session below, I outlined in blue the input command lines, and in orange, the important values returned by TraxCloud (because used in later requests), which should be different in your case.

4.1.Upload an audio file

Tip: use the tab key to let Bash auto-complete the file name. Unfortunately, this doesn't work when the "@" is present. So you can omit the "@" sign, autocomplete, and use to arrow keys to go and insert the "@" where it must be.

```
$ curla -X POST "https://trax.audionamix.com/api/v1/audiofile/" -F file=@Tell\ It\ Like\ It\ Is.wav
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
```

```

* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> POST /api/v1/audiofile/ HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
> Content-Length: 32317912
> Expect: 100-continue
> Content-Type: multipart/form-data; boundary=-----faa37e800858f120
>
< HTTP/1.1 100 Continue
< HTTP/1.1 201 CREATED
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:23:56 GMT
< Location: https://trax.audionamix.com/api/v1/audiofile/156662/
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 269
< Connection: keep-alive
<
* Connection #0 to host trax.audionamix.com left intact
{"channel_number": null, "duration": -1.0, "file": "https://
d3ijo692rza5gd.cloudfront.net/audionamix-vega-audiofiles/
Tell_It_Like_It_Is_XpRTHIV.wav", "id": 156662, "is_valid": true, "original_file_type":
null, "resource_uri": "/api/v1/audiofile/156662/", "valid": true}

```

4.2. Start separation

```

$ curla -X GET "https://trax.audionamix.com/api/v1/separation/?file_id=156662"
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/separation/?file_id=156662 HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Cache-Control: no-cache
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:25:23 GMT
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 125
< Connection: keep-alive
<
* Connection #0 to host trax.audionamix.com left intact
{"extracted_file_id": 156666, "id": 67128, "input_file_id": 156662, "resource_uri": "/
api/v1/separation/67128/", "status": 1}

```

4.3. Check separation status

```

$ curla -X GET https://trax.audionamix.com/api/v1/separation/67128/
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/separation/67128/ HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>

```

```
< HTTP/1.1 200 OK
< Cache-Control: no-cache
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:25:42 GMT
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 126
< Connection: keep-alive
<
* Connection #0 to host trax.audionamix.com left intact
{"extracted_file_id": 156666, "id": 67128, "input_file_id": 156662, "resource_uri": "/api/v1/separation/67128/", "status": 12}
```

We see the separation is currently at 12%. We wait a bit and try again:

```
$ curl -X GET https://trax.audionamix.com/api/v1/separation/67128/
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/separation/67128/ HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Cache-Control: no-cache
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:26:19 GMT
< Server: gunicorn/17.5
< Vary: Accept
< transfer-encoding: chunked
< Connection: keep-alive
<
* Connection #0 to host trax.audionamix.com left intact
{"extracted_file_id": 156666, "id": 67128, "input_file_id": 156662, "resource_uri": "/api/v1/separation/67128/", "status": 52}
```

We see the separation is currently at 52%. We wait a bit a try again:

```
$ curl -X GET https://trax.audionamix.com/api/v1/separation/67128/
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/separation/67128/ HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Cache-Control: no-cache
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:31:13 GMT
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 127
< Connection: keep-alive
<
* Connection #0 to host trax.audionamix.com left intact
```



```
{"extracted_file_id": 156666, "id": 67128, "input_file_id": 156662, "resource_uri": "/api/v1/separation/67128/", "status": 100}
```

Now we see the separation is done, and the extracted file identifier is 156666.

4.4. Download extracted audio file

We are going to download the resulting audio file to the file named “extracted.wav”:

```

$ curl -X GET "https://trax.audionamix.com/api/v1/audiofile/?pk=156666" -L -o
extracted.wav
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
   0      0     0     0         0      0      0      0      0  0*    Trying
50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
   0      0     0     0         0      0      0      0      0  0* TLS 1.0
connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/audiofile/?pk=156666 HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
   0      0     0     0         0      0      0      0      0  0< HTTP/1.1
302 FOUND
< Content-Type: text/html; charset=utf-8
< Date: Wed, 27 Jan 2016 17:31:46 GMT
< Location: https://d3ijo692rza5gd.cloudfront.net/audionamix-vega-audiofiles/
Tell_It_Like_It_Is_XpRTHIV_extracted.wav
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 0
< Connection: keep-alive
<
   0      0     0     0         0      0      0      0      0  0
* Connection #0 to host trax.audionamix.com left intact
* Issue another request to this URL: 'https://d3ijo692rza5gd.cloudfront.net/audionamix-
vega-audiofiles/Tell_It_Like_It_Is_XpRTHIV_extracted.wav'
* Trying 54.230.62.122...
* Connected to d3ijo692rza5gd.cloudfront.net (54.230.62.122) port 443 (#1)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
* Server certificate: *.cloudfront.net
* Server certificate: Symantec Class 3 Secure Server CA - G4
* Server certificate: VeriSign Class 3 Public Primary Certification Authority - G5
> GET /audionamix-vega-audiofiles/Tell_It_Like_It_Is_XpRTHIV_extracted.wav HTTP/1.1
> Host: d3ijo692rza5gd.cloudfront.net
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Content-Type: audio/x-wav
< Content-Length: 32316524
< Connection: keep-alive
< Date: Wed, 27 Jan 2016 17:31:48 GMT
< Last-Modified: Wed, 27 Jan 2016 17:26:50 GMT
< ETag: "bfff19cd7e94ff5838be489266e82c8a1"
< Accept-Ranges: bytes
< Server: AmazonS3
< X-Cache: Miss from cloudfront
< Via: 1.1 a62788af052971abc00f8142b8edd67e.cloudfront.net (CloudFront)
< X-Amz-Cf-Id: nvAcZ__p5wdZ2FvfLyrC7tA4sx3BYJY7EKI95n-_lwasmwQd26icBg==
<
   0 30.8M   0      0     0         0      0      0      0      0  0{ [16384
bytes data]

```

```
100 30.8M 100 30.8M 0 0 406k 0 0:01:17 0:01:17 --:--:-- 374k
* Connection #1 to host d3ijo692rza5gd.cloudfront.net left intact
```

We can see curl performed the redirection.

We are now going to do another separation on the same file, this time with preanalysis and annotation:

4.5. Start pre-analysis

```
$ curl -X GET "https://trax.audionamix.com/api/v1/preanalysis/start/?file_id=156662"
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/preanalysis/start/?file_id=156662 HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Cache-Control: no-cache
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:35:40 GMT
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 90
< Connection: keep-alive
<
* Connection #0 to host trax.audionamix.com left intact
{"id": 875, "is_finished": false, "resource_uri": "/api/v1/preanalysis/875/", "status": 0}
```

4.6. Check pre-analysis status

```
$ curl -X GET "https://trax.audionamix.com/api/v1/preanalysis/875/"
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/preanalysis/875/ HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Cache-Control: no-cache
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:36:12 GMT
< Server: gunicorn/17.5
< Vary: Accept
< transfer-encoding: chunked
< Connection: keep-alive
<
* Connection #0 to host trax.audionamix.com left intact
{"id": 875, "is_finished": true, "resource_uri": "/api/v1/preanalysis/875/", "status": 100}
```

The preanalysis is done, we now download the annotation file:

4.7. Download annotation file

```

$ curl -X GET "https://trax.audionamix.com/api/v1/preanalysis/875/result/" -L -o
annotation.json
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
  0     0     0     0     0     0     0     0     0  --:--:-- --:--:-- --:--:--    0*   Trying
50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
  0     0     0     0     0     0     0     0  --:--:-- --:--:-- --:--:--    0> GET /
api/v1/preanalysis/875/result/ HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 302 FOUND
< Content-Type: text/html; charset=utf-8
< Date: Wed, 27 Jan 2016 17:36:58 GMT
< Location: https://d3ijo692rza5gd.cloudfront.net/audionamix-vega-preanalysis-output/
output_pitch_trax_970nHqr.json
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 0
< Connection: keep-alive
<
  0     0     0     0     0     0     0     0  --:--:-- --:--:-- --:--:--    0
* Connection #0 to host trax.audionamix.com left intact
* Issue another request to this URL: 'https://d3ijo692rza5gd.cloudfront.net/audionamix-
vega-preanalysis-output/output_pitch_trax_970nHqr.json'
* Trying 54.230.14.150...
* Connected to d3ijo692rza5gd.cloudfront.net (54.230.14.150) port 443 (#1)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
* Server certificate: *.cloudfront.net
* Server certificate: Symantec Class 3 Secure Server CA - G4
* Server certificate: VeriSign Class 3 Public Primary Certification Authority - G5
> GET /audionamix-vega-preanalysis-output/output_pitch_trax_970nHqr.json HTTP/1.1
> Host: d3ijo692rza5gd.cloudfront.net
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Content-Type: application/json
< Content-Length: 164843
< Connection: keep-alive
< Date: Wed, 27 Jan 2016 17:36:59 GMT
< Last-Modified: Wed, 27 Jan 2016 17:36:03 GMT
< ETag: "e35fd0e1c042c65fe3e9bd26cb437c01"
< Accept-Ranges: bytes
< Server: AmazonS3
< X-Cache: Miss from cloudfront
< Via: 1.1 fff66e2ed200b85831d4967d5bfc8548.cloudfront.net (CloudFront)
< X-Amz-Cf-Id: bIxTI6GDtj7SBeA9LeBVFm5stpFKvgwDMemew6A_9HypwHtBV7GrzQ==
<
{ [16384 bytes data]
100 160k 100 160k    0     0 107k     0  0:00:01  0:00:01 --:--:-- 589k
* Connection #1 to host d3ijo692rza5gd.cloudfront.net left intact

```

Here again, we see curl followed the redirection. We can check our directory's content:

```

$ ls -la
total 126576
drwxr-xr-x  5 jdmuys  staff    170 Jan 27 18:36 .
drwx-----+ 13 jdmuys  staff   442 Jan 27 17:27 ..

```

```
-rw-r--r-- 1 jdmuys staff 32317698 Jan 26 16:16 Tell It Like It Is.wav
-rw-r--r-- 1 jdmuys staff 164843 Jan 27 18:36 annotation.json
-rw-r--r-- 1 jdmuys staff 32316524 Jan 27 18:33 extracted.wav
```

4.8. Upload the annotation file

We will not alter the annotation file in this sample session. So we upload it right away:

```
$ curla -X POST "https://trax.audionamix.com/api/v1/configuration/" -d @annotation.json
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> POST /api/v1/configuration/ HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
> Content-Length: 164843
> Content-Type: application/x-www-form-urlencoded
> Expect: 100-continue
>
< HTTP/1.1 100 Continue
* We are completely uploaded and fine
< HTTP/1.1 201 CREATED
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:37:37 GMT
< Location: https://trax.audionamix.com/api/v1/configuration/167012/
< Server: gunicorn/17.5
< Vary: Accept
< transfer-encoding: chunked
< Connection: keep-alive
<
{"big_dipper": true, "breathiness": false, "consonants_activity": "",
"extra_parameters_patch": "", "high_quality": false, "id": 167012, "maximum_frequency":
-1.0, "minimum_frequency": -1.0, "panning_vector": null, "pitch_annotation": "[[2.192,
239.8921132], [2.208, 239.8921132], [2.224, 236.4528277], [2.24, 236.4528277], [2.256,
239.8921132], [2.272, 243.3814244], [2.288, 246.9214887], [2.304, 246
4280035]], [[177.68, 223.1817502], [177.696, 223.1817502], [177.712, 219.9820378],
[177.728, 219.9820378], [177.744, 223.1817502], [177.76, 226.4280035], [177.776,
226.4280035], [177.792, 226.4280035], [177.808, 226.4280035], [177.824, 223.1817502],
[177.84, 219.9820378], [177.856, 216.828199], [177.872, 223.1817502], [177.888,
226.4280035], [177.904, 226.4280035], [177.92, 226.4280035]]]", "preference_name": null,
"processed_segments": null, "rectangle_annotation": "", "resource_uri": "/api/v1/
configuration/167012/", "reverb_time": 0.0, "voice_activity": null}

<snip>

4280035]], [[177.68, 223.1817502], [177.696, 223.1817502], [177.712, 219.9820378],
[177.728, 219.9820378], [177.744, 223.1817502], [177.76, 226.4280035], [177.776,
226.4280035], [177.792, 226.4280035], [177.808, 226.4280035], [177.824, 223.1817502],
[177.84, 219.9820378], [177.856, 216.828199], [177.872, 223.1817502], [177.888,
226.4280035], [177.904, 226.4280035], [177.92, 226.4280035]]]", "preference_name": null,
"processed_segments": null, "rectangle_annotation": "", "resource_uri": "/api/v1/
configuration/167012/", "reverb_time": 0.0, "voice_activity": null}
```

Since we have the “-v” parameter in the curla alias, curl shows the full body of the request, which contains the full content of our annotation file. I show only the few first and last lines here. We notice the parameters that you can alter. We note also the identifier of the annotation.

4.9. Start configured separation

```
$ curla -X GET "https://trax.audionamix.com/api/v1/separation/?
file_id=156662&config_id=167012"
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/separation/?file_id=156662&config_id=167012 HTTP/1.1
> Host: trax.audionamix.com
```

```

> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Cache-Control: no-cache
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:39:38 GMT
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 153
< Connection: keep-alive
<
* Connection #0 to host trax.audionamix.com left intact
{"configuration_id": 167012, "extracted_file_id": 156694, "id": 67138, "input_file_id": 156662, "resource_uri": "/api/v1/separation/67138/", "status": 1}

```

We note that we use the same audio file identifier as in the first separation.

4.10. Check separation status

```

$ curla -X GET https://trax.audionamix.com/api/v1/separation/67138/
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/separation/67138/ HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Cache-Control: no-cache
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:40:39 GMT
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 154
< Connection: keep-alive
<
* Connection #0 to host trax.audionamix.com left intact
{"configuration_id": 167012, "extracted_file_id": 156694, "id": 67138, "input_file_id": 156662, "resource_uri": "/api/v1/separation/67138/", "status": 90}

```

almost done...

```

$ curla -X GET https://trax.audionamix.com/api/v1/separation/67138/
* Trying 50.19.252.71...
* Connected to trax.audionamix.com (50.19.252.71) port 443 (#0)
* TLS 1.0 connection using TLS_DHE_RSA_WITH_AES_256_CBC_SHA
* Server certificate: *.audionamix.com
* Server certificate: RapidSSL SHA256 CA - G3
* Server certificate: GeoTrust Global CA
> GET /api/v1/separation/67138/ HTTP/1.1
> Host: trax.audionamix.com
> User-Agent: curl/7.43.0
> Accept: */*
> Authorization: ApiKey amazon:ec9cce9585c37ed7b8173c7118c7c9f5efba08c7
>
< HTTP/1.1 200 OK
< Cache-Control: no-cache
< Content-Type: application/json
< Date: Wed, 27 Jan 2016 17:40:52 GMT
< Server: gunicorn/17.5
< Vary: Accept
< Content-Length: 155

```



```

< Accept-Ranges: bytes
< Server: AmazonS3
< X-Cache: Miss from cloudfront
< Via: 1.1 a19c66da9b402e0bee3fd29619661850.cloudfront.net (CloudFront)
< X-Amz-Cf-Id: 5AQoRsnZUo_XIG-K33syedC3AmZVcSG77xLwsz70q53XQgaUPkTJgg==
<
  0 30.8M    0      0      0      0      0      0 --:--:--  0:00:03 --:--:--    0{ [16384
bytes data]
100 30.8M 100 30.8M    0      0 399k      0 0:01:18 0:01:18 --:--:-- 383k
* Connection #1 to host d3ijo692rza5gd.cloudfront.net left intact

```

And we can check the final content of our directory:

```

$ ls -la
total 190064
drwxr-xr-x  7 jdmuys  staff      238 Jan 27 18:46 .
drwx-----+ 13 jdmuys  staff      442 Jan 27 17:27 ..
-rw-r--r--   1 jdmuys  staff 32317698 Jan 26 16:16 Tell It Like It Is.wav
-rw-r--r--@   1 jdmuys  staff  186307 Jan 27 18:46 Terminal Saved Output.txt
-rw-r--r--   1 jdmuys  staff  164843 Jan 27 18:36 annotation.json
-rw-r--r--   1 jdmuys  staff 32316524 Jan 27 18:33 extracted.wav
-rw-r--r--   1 jdmuys  staff 32316524 Jan 27 18:42 result2.wav

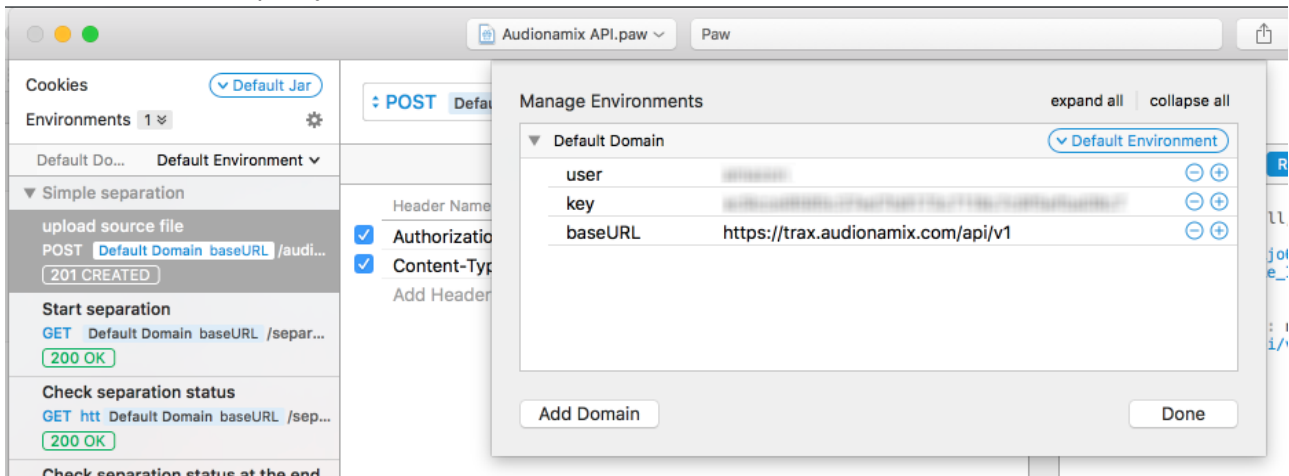
```

5. Sample session using Paw

Paw is an easy to use yet rather powerful REST client for the Macintosh (<https://luckymarmot.com/paw>). You can recreate the session from scratch using this document, or use the provided Paw document.

5.1. Set environment

While we could type everything out, Paw lets us use “variable data” instead of explicitly typed literal strings. One such source of string values is the environment. We define three environment variables: the base URL, the user id, and the api key:

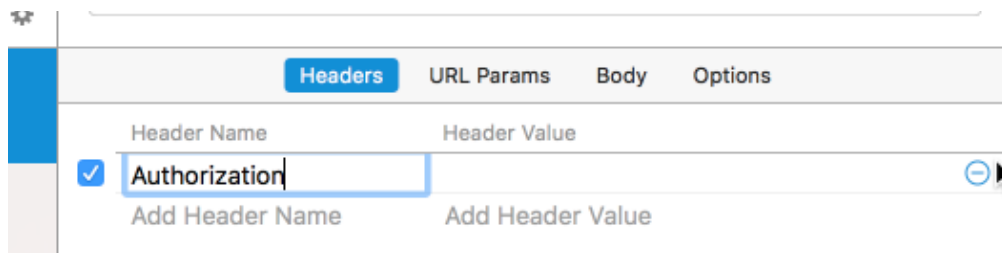


Similarly, another source of string values is the response of one request, so that we can feed it automatically in the next.

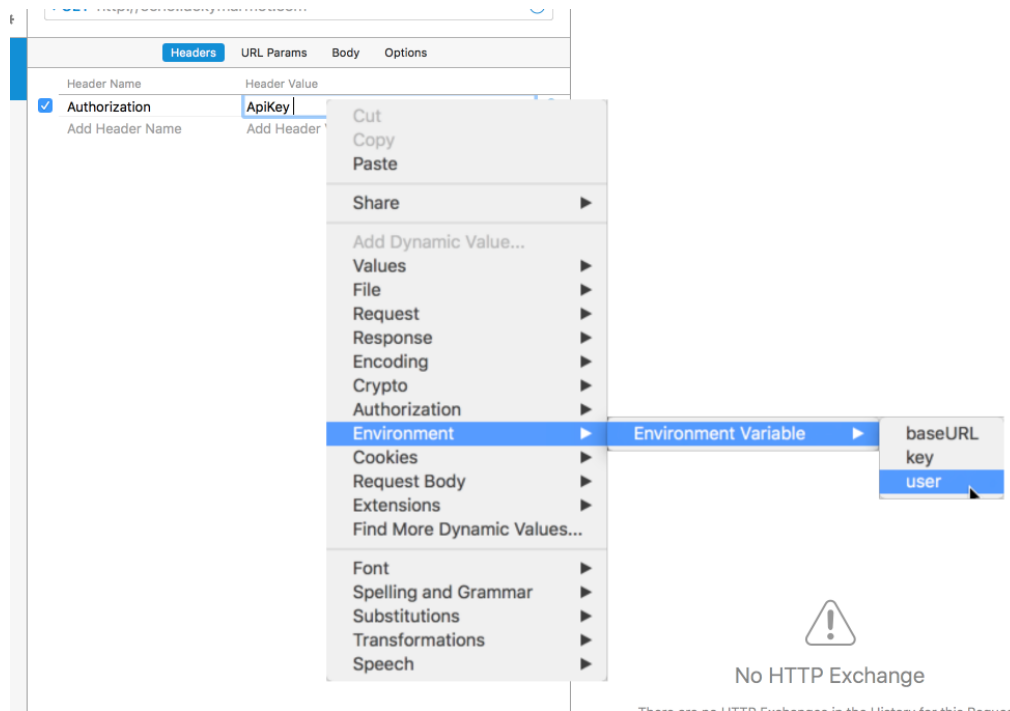
5.2. Define headers

We are going to define the only required header, the **Authorization** header. This must be done for every request. Once you’ve done it on the first request, you can duplicate that request as a starting point for the next. The duplicate will have the headers all set up.

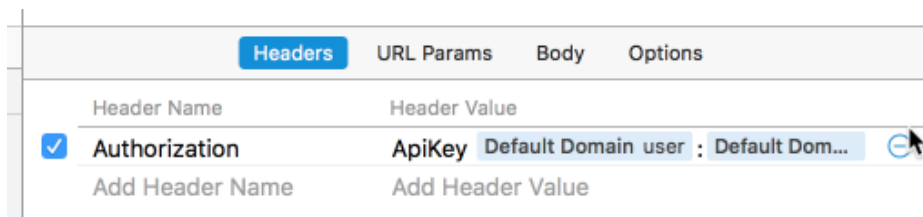
In the Headers tab of the Request pane (left pane), type “Authorization” as the header name:



For the header value, type the invariable part: “ApiKey” followed by a space. Then right-click to insert the **user** environment variable from the hierarchical menu:

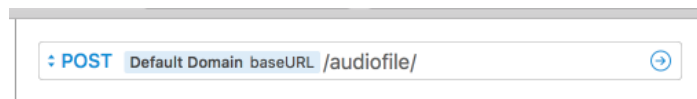


Then type the “:” separator, and insert the **key** environment variable in the same way, to get:

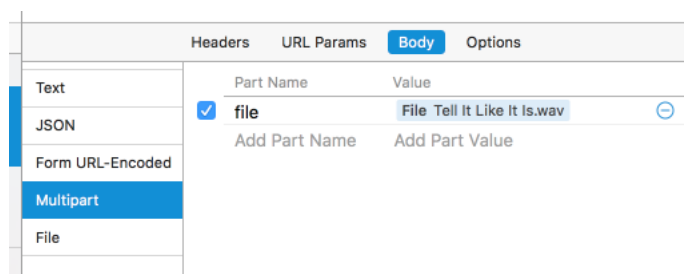


5.3. Upload the audio file

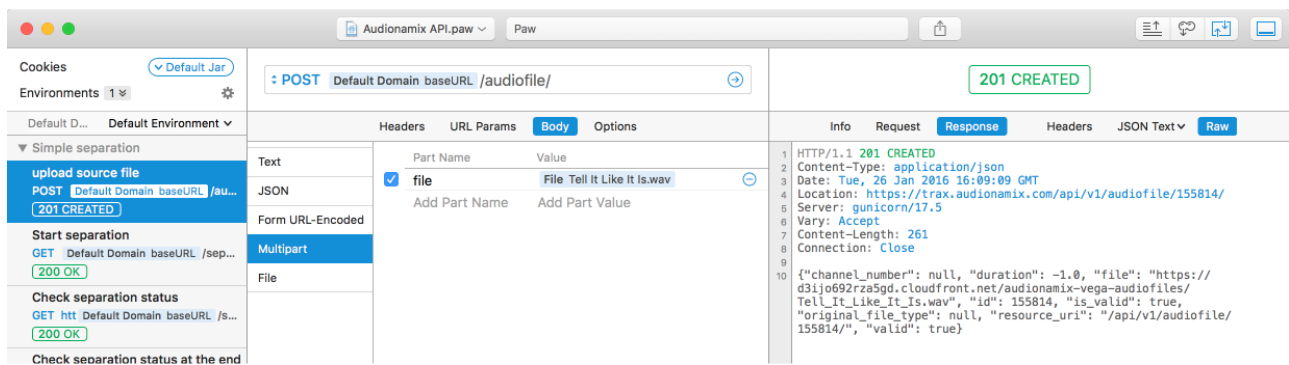
Enter the URL (using the baseUrl environment variable) and select the POST method:



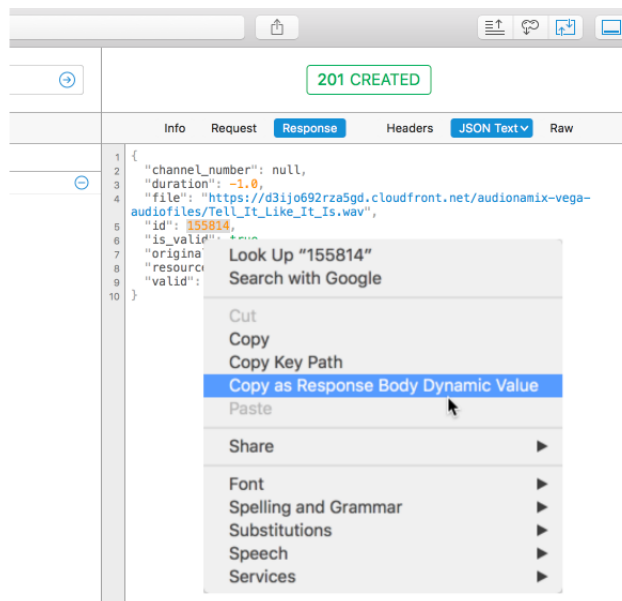
Note the arrow at the right end of the URL area. It is used to execute the request. But before that, we need to set up the body. Select the Body pan, and Multipart as the body type. Add a single part, named “file”, and for the value, right-click and select “file” as the value. Browse to your audio file, which must be a .wav file and select it. You should get something similar to:



Now you can execute the request with the little arrow. The right pane shows either the built request or the response. Here is my response, in the raw format:

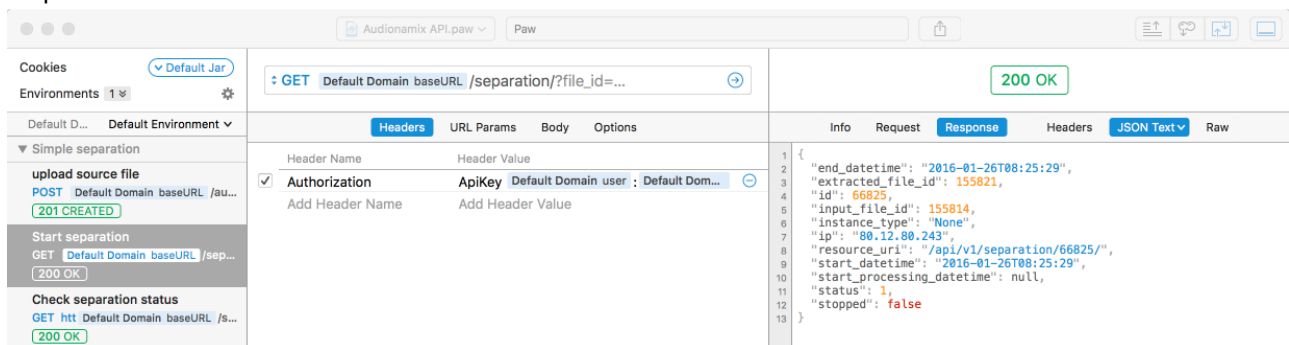


You can also select the JSON or JSON text format to display the response. In either of those, you can right click on any part of the JSON data, and copy it as dynamic value, to paste later in the next request (here the audio file identifier):



5.4. Start the separation

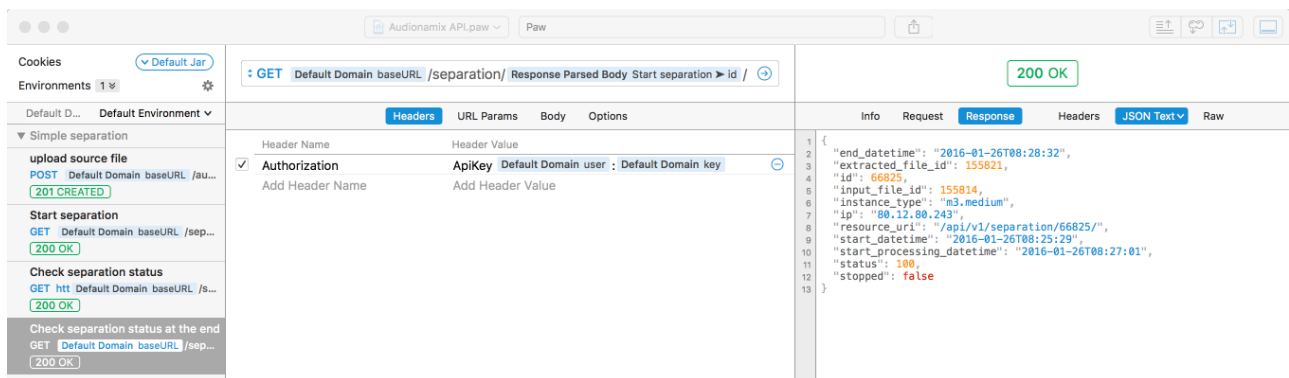
Create, in the same Paw document, a new request, to start the separation. Here it is, together with the response:



Similarly, copy the separation id as a dynamic value to paste in the next request.

5.5. Check the separation status

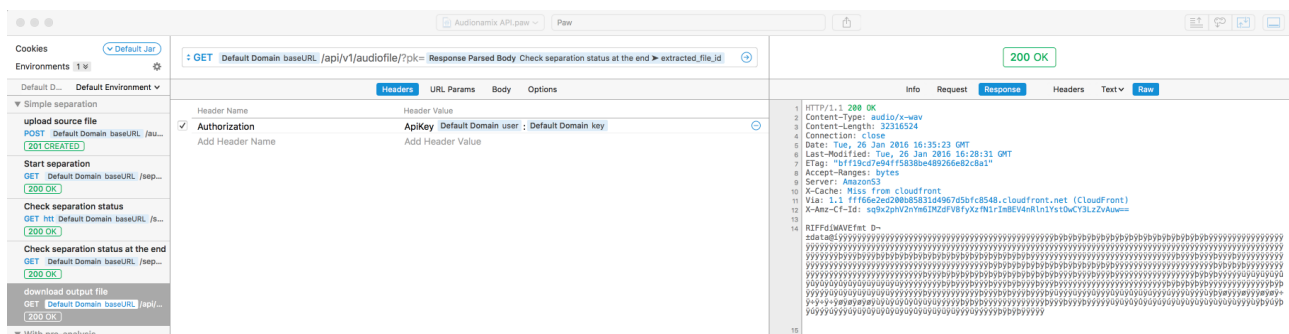
You can now guess the next request:



Copy the `extracted_file_id` as a dynamic value.

5.6. Download the resulting file

The next request, to download the resulting audio file is:



The body of the response contains the audio file. To save it on disk, go to the File menu, and select Export > Response Body.

The provided .paw document also contains the full set of requests to perform a separation with preanalysis.

6. API Reference

Note this section is not complete.

This section lists all the available resources that the API understands, what are the different way to manipulate them, the different arguments with their types and effects, the different possible HTTP results, the possible responses including a full description of the returned JSONs.

The base URI is never repeated: URI are stated relative to the base URI. The HTTP protocol is always HTTPS:

Protocol	HTTPS
Base URL	https://trax.audionamix.com/api/v1/

6.1.Template

URLs for resource resource

Method	URI	Parameters	Usage
GET	resource/schema/	none	to get the schema of resource
GET	resource/	none	to list all the resources
GET	resource/id/	none	to retrieve the resource with the given id

6.1.1.GET resource/schema/

This call returns the schema of the resource **resource**. The schema is a flat JSON object.

TBD