



Adobe Advertising APIs

Version – 2.0
Date Published - 06/22/2018
Status - Current

Overview

Adobe Advertising is the leading independent software platform that enables brands and agencies to plan, buy, measure, and optimize their global advertising.

The Adobe Advertising APIs enable you to build on the Adobe Advertising platform and leverage our reporting and capabilities in your applications. The Adobe Advertising APIs are all accessed via REST invocations and return results in JSON format.

Table of Contents:

[Overview](#)

[Adobe Advertising API Standard Information](#)

[Endpoint](#)

[Operations](#)

[Encoding](#)

[Authentication](#)

[Metadata Attributes](#)

[Date & Time](#)

[Rate Limiting](#)

[Error Codes and Responses](#)

[Adobe Advertising APIs](#)

[Audience Segment API](#)

[Metadata API](#)

[Reporting API](#)

[Trafficking API](#)

[Calling Production APIs Using Apiary](#)

[Troubleshooting](#)

[Support](#)

[Appendix A - Collections and Statistics](#)

[Collections](#)

[Statistics](#)

[Appendix B - Supported Time Zones](#)

[Appendix C - Create/Update Segment & Accounts](#)

[Appendix D - Using Postman to get an Auth Token](#)

Adobe Advertising API Standard Information

Endpoint

The base URL for the Adobe Advertising APIs is: <https://api.tubemogul.com/>

Note: Interactions with the Adobe Advertising API are supported only via HTTPS.



Operations

The Adobe Advertising APIs perform operations using standard HTTP verbs: GET, POST, PUT, and DELETE.

Verb	Used For
GET	Retrieve a resource (or set of resources)
POST	Create a resource
PUT	Update a resource (full replacement)
DELETE	Delete a resource

Encoding

Unless otherwise noted, all API interactions should be made using HTTPS requests with a `content-type` header of `application/json`. All parameters sent in `application/json` requests must conform to the JSON data interchange format, as described in [RFC 7159](#).

Authentication

Authentication methodology: The Adobe Advertising APIs use the [OAuth 2.0](#) standard to authenticate all requests. Your application plays the role of Client and Resource Owner, and the Adobe Advertising APIs play the roles of the Authorization Server and Resource Server. In short, your application is considered the owner of the data, rather than providing access to your user's data that is stored in Adobe Advertising. See the [OAuth 2.0 Roles](#) section for more details.

The only grant authentication method supported by the Adobe Advertising APIs is the [Client Credentials Grant](#) method.

Authentication process:

To use the Adobe Advertising APIs, you'll need a token that you send with every call. To retrieve a token please follow the steps below:

1. Every customer is provided a User ID (six digit account) and a API that you will use to authenticate to the Adobe Advertising APIs. If you do not know your User ID and Secret, please contact your Account Manager and they will provide your User ID and API Secret Key.



Note: Do not share either your User ID or API Secret with anyone, these resources could be used to access your data.

2. The User ID and API Secret credentials first need to be encoded to Base64, using the UTF-8 character set, before you can use them. To encode the credentials, navigate to <https://www.base64encode.org/> and enter the credentials in the form of `user_id:api_secret`. There are other methods of encoding the string, we provide this one just as a suggestion.

For example, a User ID 123456 and API Secret Key

`rhtpOwMjldQDvrxQVEpF7eVqw8QjN3Co`. Would look like this

`123456:rhtpOwMjldQDvrxQVEpF7eVqw8QjN3Co` and it would encode to create the string `MTIzNDU2OnJodHBpd01qbGRRRHZyeFFWRXBGN2VWcXc4UWpOM0Nv`.

3. Use the string you created to form the HTTP POST event. Using the example we created previously, you would send the following authorization request to `https://api.tubemogul.com/oauth/token`

```
POST /oauth/token HTTP/1.1
Host: https://api.tubemogul.com
Authorization: Basic
MTIzNDU2OnJodHBpd01qbGRRRHZyeFFWRXBGN2VWcXc4UWpOM0Nv
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded

grant_type=client_credentials
```

Please note the `Authorization`, `Cache-Control`, and `Content-Type` headers. The `Cache-Control` and `Content-Type` headers will always be the values identified in the example above, while `Authorization` will change depending on your credentials. The body of the POST should always contain `grant_type=client_credentials`.

4. The Adobe Advertising API will then respond with an access token. An example is below.

Note: These Tokens are only valid for one hour. After one hour a new token is required.

```
HTTP/1.1 200 OK
Content-Type: application/json;charset=UTF-8
Cache-Control: no-store
```

```
Pragma: no-cache

{
  "account_id":xxxxxx,
  "token":"2YotnFZFEjrlzCsicMWpAA",
  "scope":"login_mode:self",
  "token_type":"bearer",
  "expire_time":1455915931,
  "expires_in":3600
}
```

5. Pass the token you received as the `Authorization` header to access API resources. The `Authorization` header will be in the format of `'Bearer {token}'` as shown below.

```
GET /v1/trafficking/campaigns?limit=10&offset=0 HTTP/1.1
Host: https://api.tubemogul.com
Authorization: Bearer 2YotnFZFEjrlzCsicMWpAA
```

See [Appendix D](#) for an overview of how you can use Postman to get an Auth Token.

Metadata Attributes

We use @ notation in front of JSON name values to indicate metadata attributes:

Attribute	Description
@uri	A URI that identifies the resource.
@type	The type of the object being represented. Current values for @type include: collection, time_series, advertiser, campaign, placement, ad, ad_type, video.

Metadata attributes are provided in instances where the information is applicable; however, we do not provide this information in all cases. E.g. a collection where each item in the collection isn't individually accessible via its own URI.

Note: Metadata attributes cannot be directly accessed via dot-notation in Javascript. They need to be accessed using array nomenclature, e.g. `campaign["@uri"]`.

Additional information about how the Adobe Advertising APIs collect and aggregate data can be found in [Appendix A](#).

Date & Time

All date & time values are returned in [ISO-8601](#) specified format.

The timezone offset provided in the response will match the time zone provided for data consistency. Currently, only day-level granularity is supported, partial days will be truncated to the start of that day. Information about supported time zones can be found in [Appendix B](#).

Rate Limiting

Response rate from the Adobe Advertising APIs will be subject to the terms of your account. Queries in excess of that rate will return an error.

Error Codes and Responses

Error responses contain a JSON payload with a readable description of the error. A sample error response is:

```
HTTP/1.1 401 Unauthorized
  Content-Type: application/json;charset=UTF-8
  Cache-Control: no-store
  Pragma: no-cache

{
  "error_message": "Invalid user credentials supplied."
}
```

HTTP Response Code	Reason
400 (Bad Response)	Validation failure due to a malformed request.
401 (Unauthorized)	Authorization failure due to invalid credentials.
403 (Forbidden)	Attempted to access a resource that does not belong to your account.
500 (Internal Error)	Miscellaneous server error.

Adobe Advertising APIs

The Adobe Advertising APIs enable you to integrate the Adobe Advertising platform into your own platform or application. The APIs below offer different interactions with the Adobe Advertising platform. Click on each link to open detailed API documentation.

Audience Segment API

<http://docs.tmaudiencev1.apiary.io/#>

The Audience Segment methods support a real-time mapping of audience segments into a Adobe Advertising platform accounts. These methods support the management of these audience segments and their metadata. For information on creating the segment in the proper account, please see [Appendix C](#).

The Audience Segment API is currently in closed beta. If you'd like to be considered for the beta, please send a message to Grp-Adobe Advertising-IntegrationSupport@adobe.com with your request.

Information about uploading data to the Adobe Advertising Platform can be found in the [Bulk Ingestion Services](#) document.

Metadata API

<http://docs.tmmetadatav1.apiary.io/#>

The Metadata API returns general metadata information about entities (e.g. Time Zone and Product Categories) in the Adobe Advertising platform.

Reporting API

<https://tmreportingv2.docs.apiary.io/#>

The Reporting API returns metrics about different entities in the Adobe Advertising platform. This includes metrics about Campaigns, ADs, Advertisers, Placements, and more.

Trafficking API

<http://docs.tmtraffickingv1.apiary.io/#>

The Trafficking API enables the setup, modification, and deletion of advertisers, campaigns, placements, and packages.

The Trafficking API is currently in closed beta. If you'd like to be considered for the beta, please send a message to Grp-TubeMogul-IntegrationSupport@adobe.com with your request.

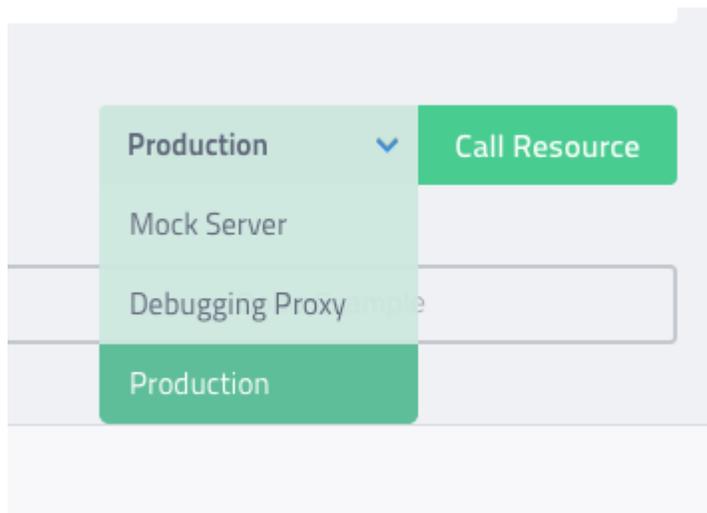
Calling Production APIs Using Apiary

Note: The functionality in this section is provided to allow a demonstration of how the Adobe Advertising APIs work. Apiary is not a tool that should be used at any point of an integration.

The Apiary tool that Adobe Advertising uses to document our APIs enables you to test our APIs and get actual responses. We recommend only performing GET actions with this service, as you will be querying your own production data.

To make production calls using Apiary, follow the steps below:

1. In the API documentation, select a resource that you would like to call.
2. Click “Try” or “Switch to Console” to open the correct interface.
3. Select “Production” from the resource selector.



4. Under the “URI Parameters” tab, enter the information that you would like to pass in your request (if applicable). There may be required parameters that you must pass. Required vs optional parameters can be identified in the API Example (dark background).

Product Categories / Get All Product Categories

GET https://api.tubemogul.com/v1/metadata/product_categories?product_category_name='cars'

URI Parameters Headers

product_category_name 'cars'

5. Under the “Headers” tab, enter your Bearer Token as the Authorization header. Details about the Authentication process can be found in the [Authentication](#) section.

Want to get a token to use in the Apiary docs? See [Appendix D](#) for a walkthrough of how to use Postman to get an Auth Token.

Product Categories / Get All Product Categories

GET https://api.tubemogul.com/v1/metadata/product_categories?product_category_name='cars'

URI Parameters Headers

Authorization Bearer 2YotnFZFEjr1zCsicMWpAA

Add a new header

6. There may be a third tab “Body” that will be present for POST or PUT operations. This is the content that you would like to write, or overwrite, to the database. (We do not recommend performing these operations in this interface.)

Campaigns / Update a Campaign

PUT <https://api.tubemogul.com/v3/trafficking/campaigns/'123'>

URI Parameters	Body	Headers
	<pre>{ "advertiser_id": "1234", "campaign_name": "my first campaign", "start_time": "2015-08-01", "end_time": "2015-12-31", "budget_cap": { "metric": "total_fees", "amount": "100000", "interval": "flight"}, "other_caps": [{ "metric": "total_fees", "amount": "20000", "interval": "month" }], "timezone_id": "15", "timezone_use_dst": true, "customer_order_number": "123456789", "third_party_services": ["nielsen_dar", "comscore_vce"], "frequency_cap": 10, }</pre>	

7. Once you've entered your token and desired information, click "Call Resource" to submit the call.

We recommend only performing GET operations in the documentation as you are querying production data. The API interface in this documentation should only be use for exploratory and testing purposes.

Note: You'll only have access to data that your User ID has been granted access to, you will get an error if you attempt to access other data:

```
{
  "errorMessage": "account is not authorized for adId: xxxxx",
  "statusCode": 403
}
```

Please contact support if you are believe that you are seeing this response in error.

Troubleshooting

Before submitting a support request, please review some common troubleshooting steps to help diagnose the issue on your own.

- 1) Check the error code and message the you're getting.
 - a) A 404 could mean you have an error in your URI.
 - b) A 401 and "Credentials are required to access this resource" means your access credentials are invalid or expired. Access tokens are only valid for 1 hour, after that time a new one is required.
 - c) If you get an HTTP 400 error, check your request data format against the API signatures.
- 2) Check the input request and response body against the documentation. Make sure your request schema matches the URI for the method you are calling and that you have the correct right input parameters for the API.

Support

Questions or bugs/issues should be sent to Grp-TubeMogul-IntegrationSupport@adobe.com. When sending an email for a support request, please include as much information as possible, e.g:

- Your name
- Your account
- Your Role & Use Case (e.g. I'm an engineer trying to retrieve metrics from the Reporting API)
- Step-by-step process of what you were trying to do
 - Step-by-step implementation (which API/Method and how you are invoking it)
 - Expected result & actual result
- Exact cURL & response received

Please direct requests for authentication or account credentials to your account manager.

Appendix A - Collections and Statistics

Collections and statistics are used by the Adobe Advertising APIs to aggregate data into meaningful groupings.

Collections

Collections consist of an `@type` attribute equal to `collection`, a `paging` object, plus an `items` array with the actual collection items.

Collections support paging by way of the `offset` and `limit` parameters provided in each request (optional). Each successive request should have `offset` incremented by `num_items` of the previous response when `has_more_items = true`. `prev/next_page_uri` links are also provided for easy collection navigation.

Parameter	Type	Required?	Description
<code>offset</code>	integer	No	The number records into the collection from which to start the retrieval. The first item in the list is <code>offset=0</code> . Default = 0
<code>limit</code>	integer	No	The maximum number of items to be returned. Limited to 1000. Default = 50
<code>num_items</code>	integer	N/A	The number of items in this page of the collection - not the entire collection.
<code>has_more_items</code>	boolean	N/A	A boolean indicating whether the list contains more items. Even if <code>num_items = limit</code> , <code>has_more_items</code> could be true or false, depending on whether there are more items in the list.
<code>prev_page_uri</code>	integer	N/A	A URI to retrieve the previous page of results. null if at the start of the list, ie. when <code>offset = 0</code> .
<code>next_page_uri</code>	integer	N/A	A URI to retrieve the next page of results. null if at the end of the list, ie. <code>has_more_items = false</code> .

Statistics

Statistics are represented as time-series data. Navigating and manipulating time-series data is accomplished with the parameters below.

Parameter	Type	Required?	Description
<code>start_day</code>	string	Yes	The start date for the report. Results are inclusive of this day. Formats adhere to ISO-8601 , ie. YYYY-MM-DD.
<code>end_day</code>	string	Yes	The end date for the report. Results are inclusive of this day. Formats adhere to ISO-8601 , ie. YYYY-MM-DD.
<code>timezone</code>	string	No	The Olson value of the time zone for the data. Individual time zones may or may not support Daylight Savings Time (DST). For each time zone that supports DST, an equivalent will also be provided that does not use DST. Supported time zones are listed in Appendix B . Default = <code>America/New_York</code> .
<code>bucketed_by</code>	string	No	Aggregation interval for time-series reports. Valid values are: <code>'day'</code> , where results will contain a bucket for each day, and <code>'range'</code> , where results will contain a bucket for the entire provided date range. Default = <code>range</code> .

Time-series do not support the pagination parameters used for paging through collections. Rather, time-series data are navigated by simply changing the start and end days of the interval being requested, along with their timezone if desired.

Note: The maximum time range that can be queried is one year.

Each stats objects map to a distinct `@type` value. These objects are:

- `{resource_type}_report` - A container object that will contain different attributes depending on the `'resource_type'`. Typically the ID and name are returned with reporting queries.
- `{resource_type}_metrics` - The supported metrics for the resource type. All resources currently support the same metrics.
- `time_series` - If `bucketed_by=day`, then each bucket will contain an attribute named `'day'`, where the value is a single day (in YYYY-MM-DD format), and data



contains metrics totals for that day. If `bucketed_by=range`, each bucket is returned with an attribute named `range`, where the value is `start_day/end_day` in YYYY-MM-DD/YYYY-MM-DD format, containing metrics totals for the entire range.

Appendix B - Supported Time Zones

Time Zones					
timezone_id	timezone_name	olson_value	supports_dst	utc_offset_non_dst	utc_offset_dst
1	Samoa Time (SST)	Pacific/Niue	F	-11:00:00	-11:00:00
3	Aleutian Time - AK (HAST/HADT)	America/Adak	T	-10:00:00	-9:00:00
5	Hawaii Time (HAST)	Pacific/Honolulu	F	-10:00:00	-10:00:00
7	Alaska Time (AKST, AKDT)	America/Anchorage	T	-9:00:00	-8:00:00
9	Pacific Time (PST, PDT)	America/Los_Angeles	T	-8:00:00	-7:00:00
11	Mountain Time (MST, MDT)	America/Denver	T	-7:00:00	-6:00:00
13	Central Time (CST, CDT)	America/Chicago	T	-6:00:00	-5:00:00
15	Eastern Time (EST, EDT)	America/New_York	T	-5:00:00	-4:00:00
17	Atlantic Time (AST, ADT)	America/Asuncion	T	-4:00:00	-3:00:00
19	Amazon Time (AMT)	Brazil/West	F	-4:00:00	-4:00:00
21	Uruguay Time (UYT)	America/Montevideo	T	-3:00:00	-2:00:00
23	Brazil Ocean Islands	America/Noronha	F	-2:00:00	-2:00:00
25	Azores Time (AZOT, AZOST)	Atlantic/Azores	T	-1:00:00	0:00:00
27	Western European Time (WET, WEST)	Europe/Lisbon	T	0:00:00	1:00:00
29	British Time (BST)	Europe/London	T	0:00:00	1:00:00
31	North Atlantic/West Africa (GMT)	GMT	F	0:00:00	0:00:00
33	Central European Time (CET, CEST)	Europe/Berlin	T	+1:00:00	2:00:00
35	Eastern European Time (EET, EEST)	Asia/Istanbul	T	+2:00:00	3:00:00
37	Arabia Time (AST)	Asia/Riyadh	F	+3:00:00	3:00:00
39	Eastern Europe Forward Time (EEFT)	Europe/Kaliningrad	F	+3:00:00	3:00:00
41	Azerbaijan Time (AZT, AZST)	Asia/Baku	T	+4:00:00	5:00:00
43	Moscow Time (MSK)	Europe/Moscow	F	+4:00:00	4:00:00
45	Pakistan Time (PKT)	Asia/Karachi	F	+5:00:00	5:00:00

47	Central Asia	Asia/Dhaka	F	+6:00:00	6:00:00
49	Indochina Time (ICT)	Asia/Bangkok	F	+7:00:00	7:00:00
51	China Time (CST)	Asia/Chungking	F	+8:00:00	8:00:00
53	Singapore Time (SGT)	Asia/Singapore	F	+8:00:00	8:00:00
55	Australian Western Standard Time (AWST)	Australia/Perth	F	+8:00:00	8:00:00
57	Korea Standard Time (KST)	Asia/Seoul	F	+9:00:00	10:00:00
59	Japan Standard Time (JST)	Asia/Tokyo	F	+9:00:00	10:00:00
61	Asia Pacific Time	Asia/Yakutsk	F	+10:00:00	10:00:00
63	Australian Eastern Time (AEST, AEDT)	Australia/Sydney	T	+10:00:00	11:00:00
65	East Asia Time	Asia/Vladivostok	F	+11:00:00	11:00:00
67	New Zealand Standard Time (NZST, NZDT)	Pacific/Auckland	T	+12:00:00	13:00:00
69	West Samoa Time (WST, WSST)	Pacific/Apia	T	+13:00:00	14:00:00
71	Line Islands Time (LINT)	Pacific/Kiritimati	F	+14:00:00	14:00:00

Appendix C - Create/Update Segment & Accounts

When using the [Audience Segment API](#), you may need to create segments that belong to different accounts. For example, parent company (or agency) *Hooli* may manage ad user segments for its brands (or accounts) *Nucleus* and *EndFrame*. This section will describe how to create segments for the different accounts.

- 1) Authentication is done in the same manner as the [Authentication](#) section above. You'll log in with the parent company / agency account (e.g. hooli@Adobe Advertising.com) and retrieve your user ID and secret key. Follow the steps in that section to retrieve and use your token.
- 2) When calling the 'Create Segment' and 'Update Segment' method you'll need to specify which account you want to create the segment in. To do this, you'll use an 'Account Code' for each account. If you don't have the codes for your account contact Grp-TubeMogul-IntegrationSupport@adobe.com to retrieve them. Please include the account login email for each account that you would like the account code for.
3. Use the codes provided when calling the 'Create Segment' method and pass in the body as the 'account_token'.

Note: The parent account, the one used for authentication to the APIs, needs to have permission to make changes in the brand accounts. Otherwise the API will return an error.

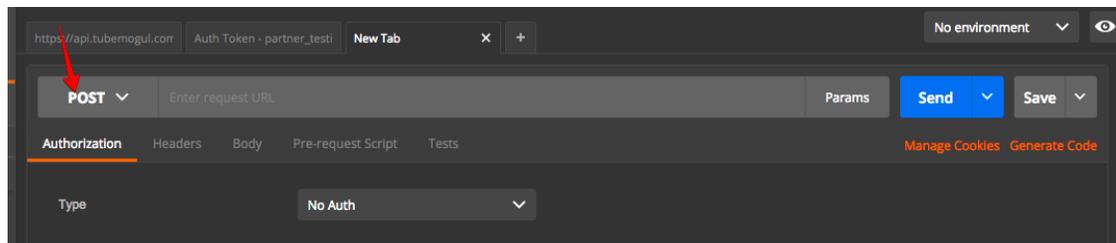
Appendix D - Using Postman to get an Auth Token

When using Apiary to test the Adobe Advertising APIs, you'll first need to grab an Auth Token as described in the [Authentication](#) section. The process to get an Auth Token can be completed many ways, using [Postman](#) is just one common method. Another method would be to use cURL in your command line.

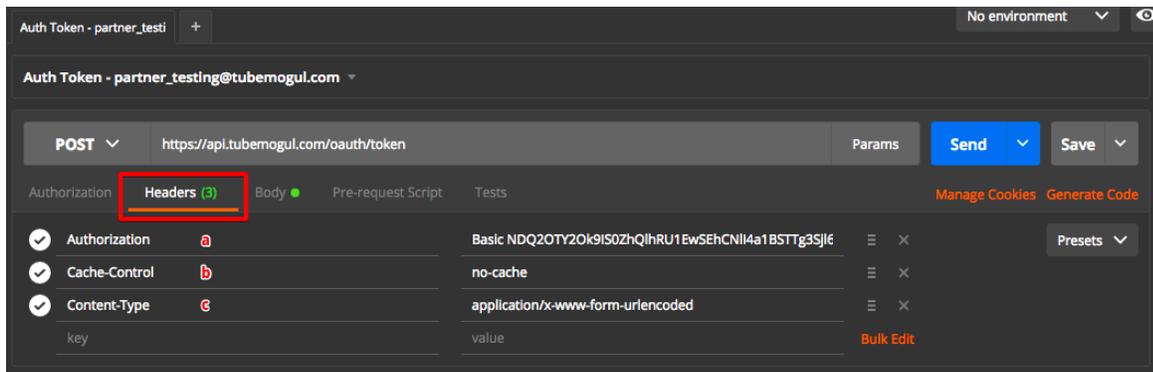
1. First you'll need to encode your credentials to create an 'Authentication' header. Go to <https://www.base64encode.org> and enter your credentials, in `user_id:api_secret` format. Hit 'Encode' to receive your string. This is part of your 'Authorization' header.

For example, a User ID 123456 and API Secret Key
`rhttpOwMjldQDvrxQVEpF7eVqw8QjN3Co`, would encode to create the string
`MTIzNDU2OnJodHBPd01qbGRRRHZyeFFWRXBGN2VWcXc4UWpOM0Nv.`

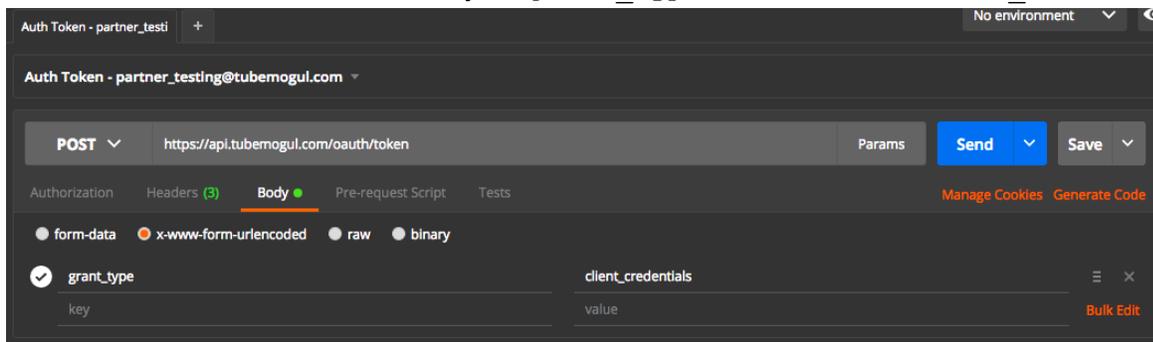
2. Launch [Postman](#) and select the HTTP operation that you'd like to invoke, in this case select POST.



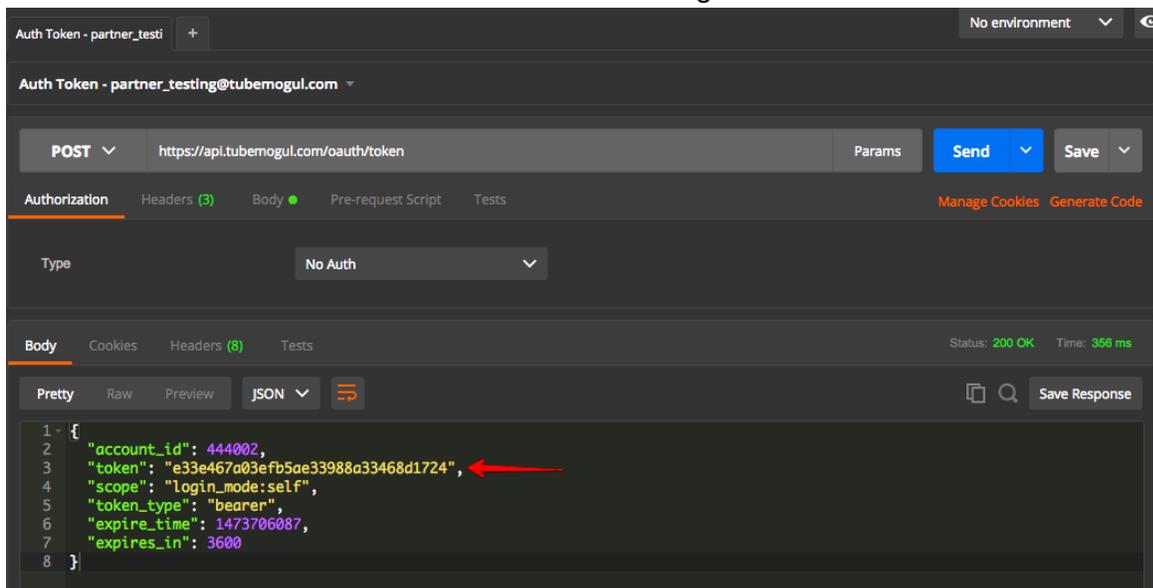
3. In the box that says "Enter request URL" input <https://api.tubemogul.com/oauth/token> as the token endpoint.
4. Select the tab labeled "Headers" and add the following:
 - a) 'key' = "Authorization" and 'value' = "Basic {encoded string}". Replace {encoded string} with the output from step 1.
 - b) 'key' = "Cache-Control" and 'value' = "no-cache".
 - c) 'key' = "Content-Type" and 'value' = "application/x-www-form-urlencoded"



5. Finally, under the tab labeled “Body” select the radio button that labeled “x-www-form-urlencoded” and enter ‘key’ = “grant_type” and ‘value’ = “client_credentials”.



6. To send the POST, click the “Send” button. Your token will be contained in the response. Use this token to access the Adobe Advertising API resources.



Note: The token you receive by following these steps is only valid for 1 hour.



Don't have the credentials required in step 1? Ask your Adobe Advertising account manager. They can retrieve them for you.

Followed the steps and having issues getting a token? Contact Grp-TubeMogul-IntegrationSupport@adobe.com.

Troubleshooting

Something go wrong? Troubleshooting for the most common error messages are below.

1. The most common error response received is:

```
1 {  
2   "error": "invalid_client",  
3   "error_description": "invalid credentials"  
4 }
```

To resolve this, please double check your encoded "Authorization" header.

2. If you receive the following:

```
1 {  
2   "error": "invalid_grant",  
3   "error_description": "grant_type must be 'client_credentials'"  
4 }
```

Check to make sure your 'grant_type' is 'client_credentials'.