

1 Campaign API Reference

The following operations are available via the API:

1. Fetch a List of Campaigns - Fetch a list of Campaigns associated with a specific partner account.
2. Fetch a Single Campaign - Fetch a single Campaign resource using the Campaign ID.
3. Create a Campaign - Create a new Campaign under a specific partner account.
4. Update a Campaign - Update an existing Campaign of a specific partner account.

1.1 Cheat Sheet



This is merely a quick reference of the available end-points. For further information please continue to the sections below.

Description	Method	Path (prefix: /backstage/api/1.0)
Fetch a list of campaigns associated with an account More info	GET	/[account-id]/campaigns/
Fetch a specific campaign associated with an account More info	GET	/[account-id]/campaigns/[campaign-id]/
Create a new campaign for an account More info	POST	/[account-id]/campaigns/
Update an existing campaign More info	POST, PUT	/[account-id]/campaigns/[campaign-id]/



Reading, creating or updating Campaigns is only possible when the [account-id] references an Account with ADVERTISER in its partner_types field. Trying to send a request with a non-Advertiser Account will result in a 404 Not Found response.

1.2 Campaign Resource

1.2.1 Fields



Required fields are fields that must be sent to the server when creating a new resource.

Read-only fields are fields that should never be sent to the server, and will appear only when *fetching* a resource.

Final fields are set once, when creating the resource, and become *read-only* afterwards.

Name	Modifier & Defaults	Type	Description
id	Read-only	String	Unique numeric ID of a Campaign (returned as a string)
advertiser_id	Read-only	String	The account_id of the Campaign's Advertiser Account
name	Required	String Max length: 200	Human-readable name for the Campaign
branding_text	Required	String Max length: 25	This text will appear below the title of each of your items
tracking_code	Default: "utm_source=taboola&utm_medium=referral"	String Max length: 255	Query-string parameter which will be added to the URL of your items, so you can track the traffic generated from Taboola.
cpc	Required	Double Must be in the allowed range of the resource's account.	Cost-per-click
daily_cap	Default: 0.0	Double Must be lower than the spending_limit 0.0 means "unlimited"	
daily_ad_delivery_model	Default: Depending on daily_cap field	String Possible values:	The Daily Ad Delivery Model (combined with the Daily Cap)


	<p>(for backward compatibility):</p> <p>If <code>daily_cap</code> exist and <code>> 0</code> - this field defaults to "STRICT"</p> <p>Otherwise it defaults to "ACCELERATED"</p>	<table><tr><td>BALANCED</td><td>The daily cap will be balanced each day of the month according to the monthly budget. Daily Cap must be empty.</td></tr><tr><td>ACCELERATED</td><td>The campaign will be delivered without a daily cap. Same as setting <code>daily_cap</code> to <code>0.0</code>.</td></tr><tr><td>STRICT</td><td>The campaign's daily cap will be a strict one determined by the <code>daily_cap</code> field. Same as setting <code>daily_cap</code> higher than <code>0.0</code>.</td></tr></table>	BALANCED	The daily cap will be balanced each day of the month according to the monthly budget. Daily Cap must be empty.	ACCELERATED	The campaign will be delivered without a daily cap. Same as setting <code>daily_cap</code> to <code>0.0</code> .	STRICT	The campaign's daily cap will be a strict one determined by the <code>daily_cap</code> field. Same as setting <code>daily_cap</code> higher than <code>0.0</code> .	defines the way the campaign will be delivered on a daily basis. Each type represents a different approach to how the campaign budget will be spent.
BALANCED	The daily cap will be balanced each day of the month according to the monthly budget. Daily Cap must be empty.								
ACCELERATED	The campaign will be delivered without a daily cap. Same as setting <code>daily_cap</code> to <code>0.0</code> .								
STRICT	The campaign's daily cap will be a strict one determined by the <code>daily_cap</code> field. Same as setting <code>daily_cap</code> higher than <code>0.0</code> .								
<code>publisher_bid_modifier</code>	Default: Publisher Bid Modifier Object with <code>values=[]</code>	Publisher Bid Modifier Object	An object representing the wanted publisher bid modifiers for this campaign						
<code>spending_limit</code>	Required	Double Must be higher than the following fields: <code>cpc</code> , <code>daily_cap</code>	The maximum amount of money a Campaign can consume.						
<code>spending_limit_model</code>	Required	String Most updated possible values are accessible via dictionary. Here's the current ones:	The Spending Limit Model defines whether the campaign will have a monthly recurring budget (which						

		<table><tr><td>MONTHLY</td><td>Monthly recurring budget</td></tr><tr><td>ENTIRE</td><td>Non recurring budget</td></tr></table>	MONTHLY	Monthly recurring budget	ENTIRE	Non recurring budget	replenishes automatically each month) or an overall budget.
MONTHLY	Monthly recurring budget						
ENTIRE	Non recurring budget						
country_targeting	Default: Targeting Object with type=ALL	Targeting Object NULL means run on all countries Possible <i>targeting</i> values: 2-letter country code as defined by ISO-3166 ; Most updated possible values are accessible via dictionary.	List of countries the campaign will target or exclude.				
sub_country_targeting	Default: Targeting Object with type=ALL	Targeting Object NULL means run on all regions / DMA's. This feature can only be used if only 1 country is used in country targeting in 'INCLUDE' mode, and the region or DMA codes must be relevant to that country. In addition, this can be used for region targeting or DMA targeting (DMA only in US) but not both at the same time. Possible <i>targeting</i> values: Most updated possible values are accessible via dictionary.	List of regions or DMA's the campaign will target or exclude.				
platform_targeting	Default: Targeting Object with type=ALL	Targeting Object (INCLUDE only) NULL means to run on all platforms Most updated possible values are accessible via dictionary. Here's the current ones:: <table><tr><td>DESK</td><td>Desktop</td></tr><tr><td>PHON</td><td>Smartphone</td></tr></table>	DESK	Desktop	PHON	Smartphone	List of platform device types (desktop / tablet / smartphone) to run the campaign on. NULL means to run on all platform device types.
DESK	Desktop						
PHON	Smartphone						

		<table><tr><td>TBLT</td><td>Tablet</td></tr></table>	TBLT	Tablet		
TBLT	Tablet					
publisher_targeting	Default: Targeting Object with <code>type=ALL</code>	Targeting Object (EXCLUDE only) NULL means to run on all publishers Possible <i>targeting</i> values: <code>account_id</code> of various Accounts	List of Partner Accounts' <code>account_id</code> to blacklist from publishing this Campaign. The <code>account_id</code> must reference an Account with <code>type=PARTNER</code> .			
comments	Default: "" (empty string)	String Max length: 1000	Campaign comments			
start_date	Final Default: <Now> OR: NULL (for non-privileged users)	Date (ISO 8601) Must be today or later.				
end_date	Default: "9999-12-31"	Date (ISO 8601) Must be later than <code>start_date</code> Can be modified only if <code>status!=EXPIRED</code>				
approval_state	Read-only Default: PENDING	String Requires certain permissions to modify Possible values: <table><tr><td>APPROVED</td></tr><tr><td>REJECTED</td></tr><tr><td>PENDING</td></tr></table>	APPROVED	REJECTED	PENDING	This field determines whether the Campaign is approved to be served. The Campaign can only be served if its <code>approval_state</code> is <code>APPROVED</code> . This field affects the <code>status</code> of the Campaign.
APPROVED						
REJECTED						
PENDING						

is_active	Default: true	Boolean	<div>If true, Campaign is active and may serve ads if its status is RUNNING.</div> <div>If false, Campaign is currently not serving ads, and its status will be PAUSED</div>																		
spent	Read-only	Double	Represents the estimated amount of money this Campaign has consumed																		
status	Read-only	<div>String Most updated possible values are accessible via dictionary. Here are the current ones:</div> <table><tr><td>RUNNING</td><td>Campaign is currently serving ads</td></tr><tr><td>PAUSED</td><td>Campaign is currently not serving ads</td></tr><tr><td>PENDING_START_DATE</td><td>Campaign is pending its start_date</td></tr><tr><td>DEPLETED_MONTHLY</td><td>Campaign has depleted its MONTHLY spending_limit</td></tr><tr><td>DEPLETED</td><td>Campaign has depleted its ENTIRE spending_limit</td></tr><tr><td>EXPIRED</td><td>Campaign's end_date has passed</td></tr><tr><td>TERMINATED</td><td>Campaign was manually terminated</td></tr><tr><td>FROZEN</td><td>Campaign's Advertiser account is Frozen</td></tr><tr><td>PENDING_APPROVAL</td><td>Campaign has approval_status set to PENDING</td></tr></table>		RUNNING	Campaign is currently serving ads	PAUSED	Campaign is currently not serving ads	PENDING_START_DATE	Campaign is pending its start_date	DEPLETED_MONTHLY	Campaign has depleted its MONTHLY spending_limit	DEPLETED	Campaign has depleted its ENTIRE spending_limit	EXPIRED	Campaign's end_date has passed	TERMINATED	Campaign was manually terminated	FROZEN	Campaign's Advertiser account is Frozen	PENDING_APPROVAL	Campaign has approval_status set to PENDING
RUNNING	Campaign is currently serving ads																				
PAUSED	Campaign is currently not serving ads																				
PENDING_START_DATE	Campaign is pending its start_date																				
DEPLETED_MONTHLY	Campaign has depleted its MONTHLY spending_limit																				
DEPLETED	Campaign has depleted its ENTIRE spending_limit																				
EXPIRED	Campaign's end_date has passed																				
TERMINATED	Campaign was manually terminated																				
FROZEN	Campaign's Advertiser account is Frozen																				
PENDING_APPROVAL	Campaign has approval_status set to PENDING																				

		REJECTED	Campaign has <code>approval_status</code> set to <code>REJECTED</code>
--	--	----------	--



Trying to set field values which do not adhere to their restrictions, will result in a `400 Bad Request` error response.

1.2.2 Campaign Spending Models

Generally speaking, Campaigns have two possible "Spending Models":

- **Entire Budget** - The `spending_limit` field of the Campaign represents the entire amount of money this Campaign can consume. This means that once it depletes this budget, the Campaign will stop running, and its `status` will be `DEPLETED`.
- **Monthly Budget** - The `spending_limit` field of the Campaign represents the amount of money this Campaign can consume *each month*. This means that once a Campaign depletes its budget, the Campaign will stop running, and its `status` will be `MONTHLY_DEPLETED`. Once a new month begins, the Campaign's budget will replenish and the Campaign will start running again.

The Spending Model type of a certain Campaign is determined by setting the value of its `spending_limit_model` field. Setting the field to `ENTIRE` will set the Spending Model of the Campaign to "Entire Budget", while `MONTHLY` will set it to "Monthly Budget".

1.2.3 Publisher Bid Modifier Object

1.2.4 Fields

Name	Modifier & Defaults	Type	Description
values	Default: <code>[]</code>	Array< Targeted Bid Modifier Object > NULL means no change to targeted bid modifiers [] means no targeted bid modifiers	An object representing the targeted publishers bid modifiers

1.2.5 Targeted Bid Modifier Object

1.2.6 Fields

Name	Modifier & Defaults	Type	Description
target	Required	String The name of the target to associated this bid boost modifier	When using in context of " Publisher Bid Modifier Object ", target is publisher name
cpc_modification	Required	Double Must be between values 0.5 - 1.5	

1.3 Targeting Field

1.3.1 Fields

Name	Modifier & Defaults	Type	Description				
type	Required	<div>String Most updated possible values are accessible via dictionary. Here are the current ones:</div> <table><tr><td>INCLUDE</td><td>Only supplied values should be included in Targeting</td></tr><tr><td>EXCLUDE</td><td>All values should be Targeted other than the supplied values.</td></tr></table>	INCLUDE	Only supplied values should be included in Targeting	EXCLUDE	All values should be Targeted other than the supplied values.	Type of targeting - Whether the listed values should be Blacklisted (EXCLUDE) or Whitelisted (INCLUDE)
INCLUDE	Only supplied values should be included in Targeting						
EXCLUDE	All values should be Targeted other than the supplied values.						

		<table><tr><td>ALL</td><td>No restrictions - target all options.</td></tr></table>	ALL	No restrictions - target all options.	
ALL	No restrictions - target all options.				
value	Required	Array<String> Possible values depend on the possible targeting values of the specific field	An array of values to target		

1.4 Reading

Reading can be performed in two ways:

1. Fetch a list of Campaigns
2. Fetch a single Campaign

1.4.1 Fetching a List

Request

To retrieve a list of an Account's Campaigns, send an HTTP *GET* request in the following form:

```
GET /backstage/api/1.0/[account-id]/campaigns/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]
```

Example Response:

200 OK

```
{
  "results": [
    {
      "id" : "1",
      "advertiser_id" : "taboola-demo-advertiser",
      "name" : "Demo Campaign",
      "tracking_code" : "taboola-track",
      "cpc" : 0.25,
      "daily_cap" : 100,
      "spending_limit" : 1000,
      "spending_limit_model": "MONTHLY",
      "country_targeting": {
        "type" : "INCLUDE",
        "value" : ["AU", "GB"]
      },
      "sub_country_targeting": null,
      "platform_targeting": {
        "type" : "INCLUDE",
```

```

        "value" : ["TBLT", "PHON"]
    },
    "publisher_targeting": null,
    "comments" : null,
    "start_date" : "2015-04-30",
    "end_date" : "2015-05-30",
    "approval_state" : "APPROVED",
    "is_active" : true,
    "spent" : 2.23,
    "status" : "RUNNING",
    "daily_ad_delivery_model" : "STRICT",
    "publisher_bid_modifier" : {
        "values": [{ "target" : "publisher1", "cpc_modification" : 1.5},
                   { "target" : "publisher2", "cpc_modification" : 0.9}
        ]
    }
},
// ... more results
]
}

```

1.4.2 Fetching a Specific Instance

Request

To retrieve a specific Campaign, send an HTTP *GET* request in the following form:

```

GET /backstage/api/1.0/[account-id]/campaigns/[campaign-id]/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]

```

The response will include a valid JSON object in its body representing the Campaign resource.

Example Response:

```

200 OK

{
  "id": "1",
  "advertiser_id" : "taboola-demo-advertiser",
  "branding_text": "Pizza",
  "name": "Demo Campaign",
  "tracking_code": "taboola-track",
  "cpc": 0.25,
  "daily_cap": 100,
  "spending_limit": 1000,
  "spending_limit_model": "MONTHLY",
  "country_targeting": {
    "type" : "INCLUDE",
    "value" : ["AU", "GB"]
  }
}

```

```

    },
    "sub_country_targeting": null,
    "platform_targeting": {
      "type" : "INCLUDE",
      "value" : ["TBLT", "PHON"]
    },
    "publisher_targeting": null,
    "comments" : null,
    "start_date": "2015-04-30",
    "end_date": "2015-05-30",
    "approval_state" : "APPROVED",
    "is_active": true,
    "spent": 0.0,
    "status": "RUNNING",
    "daily_ad_delivery_model" : "STRICT",
    "publisher_bid_modifier" : {
      "values": [{"target" : "publisher1", "cpc_modification" : 1.5},
                 {"target" : "publisher2", "cpc_modification" : 0.9}]
    }
  }
}

```

1.5 Creating

1.5.1 Request Structure

Creating a new Campaign is performed by sending an HTTP *POST* request to a URL in the following form:

```

POST /backstage/api/1.0/[account-id]/campaigns/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]

```

The request must include a JSON object representing the Campaign in its body.



The body of the request must contain *at least* all of the *required* fields. An incomplete set of fields will result in a **400 Bad Request** error response.

The response for such a request will be **200 OK** and will include the newly created campaign as a valid JSON object in its body.

1.5.2 Example: Only Required Fields

```

POST /backstage/api/1.0/taboola-demo-advertiser/campaigns/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]

```

```
{
  "name": "Demo Campaign",
  "branding_text": "Pizza",
  "cpc": 0.25,
  "spending_limit": 1000,
  "spending_limit_model": "MONTHLY"
}
```

Response:

200 OK

```
{
  "id": "123",
  "advertiser_id" : "taboola-demo-advertiser",
  "name": "Demo Campaign",
  "branding_text": "Pizza",
  "tracking_code": "",
  "cpc": 0.25,
  "daily_cap": null,
  "spending_limit": 1000,
  "spending_limit_model": "MONTHLY",
  "country_targeting": null,
  "sub_country_targeting": null,
  "platform_targeting": null,
  "publisher_targeting": null,
  "comments" : null,
  "start_date": null,
  "end_date": null,
  "approval_state" : "APPROVED",
  "is_active": true,
  "spent": 0.0,
  "status": "RUNNING",
  "daily_ad_delivery_model" : "ACCELERATED",
  "publisher_bid_modifier": {
    "values": []
  }
}
```



The response includes the read-only fields `id` and `status`.

1.5.3 Example: Full Set of Fields

```
POST /backstage/api/1.0/taboola-demo-advertiser/campaigns/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]
```

```
{
```

```

"name": "Demo Campaign",
"branding_text": "Pizza",
"tracking_code": "taboola-track",
"cpc": 0.25,
"daily_cap": 100,
"spending_limit": 1000,
"spending_limit_model": "MONTHLY",
"country_targeting": {
  "type" : "INCLUDE",
  "value" : ["AU", "GB"]
},
"sub_country_targeting": null,
"platform_targeting": {
  "type" : "INCLUDE",
  "value" : ["TBLT", "PHON"]
},
"publisher_targeting": null,
"daily_ad_delivery_model" : "STRICT",
"publisher_bid_modifier" : {
  "values": [{"target" : "publisher1", "cpc_modification" : 1.5},
             {"target" : "publisher2", "cpc_modification" : 0.9}
]
},
"comments" : null,
"start_date": "2015-04-30",
"end_date": "2015-05-30",
"approval_state" : "APPROVED",
"is_active": true
}

```

Response:

200 OK

```

{
  "id": "124",
  "advertiser_id" : "taboola-demo-advertiser",
  "name": "Demo Campaign",
  "branding_text": "Pizza",
  "tracking_code": "taboola-track",
  "cpc": 0.25,
  "daily_cap": 100,
  "spending_limit": 1000,
  "spending_limit_model": "MONTHLY",
  "country_targeting": {
    "type" : "INCLUDE",
    "value" : ["AU", "GB"]
  },
  "sub_country_targeting": null,
  "platform_targeting": {
    "type" : "INCLUDE",
    "value" : ["TBLT", "PHON"]
  },
  "publisher_targeting": null,
  "comments" : null,
  "start_date": "2015-04-30",
  "end_date": "2015-05-30",
  "approval_state" : "APPROVED",
}

```

```

    "is_active": true,
    "spent": 0.0,
    "status": "RUNNING",
    "daily_ad_delivery_model" : "STRICT",
    "publisher_bid_modifier" : {
      "values": [{"target" : "publisher1", "cpc_modification" : 1.5},
                 {"target" : "publisher2", "cpc_modification" : 0.9}
    ]
  }
}

```

1.5.4 Example: Insufficient Set of Fields

```

POST /backstage/api/1.0/taboola-demo-advertiser/campaigns/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]

```

```

{
  "name": "Demo Campaign",
  "branding_text": "Pizza",
  "spending_limit": 1000,
  "spending_limit_model": "MONTHLY",
}

```

Response:

```

400 Bad Request

```

```

{
  "http_status": 400,
  "message": "\"cpc\" field is missing."
}

```

1.6 Updating

Updating is performed by sending an HTTP *POST* or *PUT* request, which includes a JSON object representing the fields to update in its body:

```

POST /backstage/api/1.0/[account-id]/campaigns/[campaign-id]/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]

```



When updating resources, it is possible to use either the **POST** method, or the **PUT** method. The API will respect both, but the documentation only uses the **POST** approach in examples.



An update request may include only a subset of the resource's fields. Included fields with **NULL** value will be treated as if they are missing from the request.
Fields which are missing or set to null in the request will not be modified.

1.6.1 Example: Update a Single Field

```
POST /backstage/api/1.0/taboola-demo-advertiser/campaigns/124/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]
```

```
{
  "name": "Demo Campaign - Edited"
}
```

Response:

200 OK

```
{
  "id": "124",
  "advertiser_id" : "taboola-demo-advertiser",
  "name": "Demo Campaign - Edited",
  "branding_text": "Pizza",
  "tracking_code": "taboola-track",
  "cpc": 0.25,
  "daily_cap": 100,
  "spending_limit": 1000,
  "spending_limit_model": "MONTHLY",
  "country_targeting": {
    "type" : "INCLUDE",
    "value" : ["AU", "GB"]
  },
  "sub_country_targeting": null,
  "platform_targeting": {
    "type" : "INCLUDE",
    "value" : ["TBLT", "PHON"]
  },
  "publisher_targeting": null,
  "comments" : null,
  "start_date": "2015-04-30",
  "end_date": "2015-05-30",
  "approval_state" : "APPROVED",
  "is_active": true,
  "spent": 0.0,
  "status": "RUNNING",
  "daily_ad_delivery_model" : "STRICT",
  "publisher_bid_modifier" : {
    "values": [{"target" : "publisher1", "cpc_modification" : 1.5},
               {"target" : "publisher2", "cpc_modification" : 0.9}]
  }
}
```

1.6.2 Example: Updating Multiple Fields

POST /backstage/api/1.0/**taboola-demo-advertiser**/campaigns/124

Host: https://backstage.taboola.com

Authorization: Bearer **[access-token]**

```
{
  "name": "Demo Campaign",
  "branding_text": "Yummy Pizza",
  "tracking_code": "taboola-tracking-code",
  "cpc": 0.5,
  "daily_cap": 500,
  "spending_limit": 5000,
  "spending_limit_model": "ENTIRE",
  "country_targeting": {
    "type" : "INCLUDE",
    "value" : ["AU", "GB"]
  },
  "sub_country_targeting": null,
  "platform_targeting": {
    "type" : "INCLUDE",
    "value" : ["TBLT", "PHON"]
  },
  "publisher_targeting": null,
  "publisher_bid_modifier" : {
    "values": [{"target" : "publisher1", "cpc_modification" : 1.5},
               {"target" : "publisher2", "cpc_modification" : 0.9}
    ]
  },
  "comments" : null,
  "start_date": "2015-04-24",
  "end_date": "2015-05-24"
}
```

Response:

200 OK


```

{
  "id": "124",
  "advertiser_id" : "taboola-demo-advertiser",
  "name": "Demo Campaign",
  "branding_text": "Yummy Pizza",
  "tracking_code": "taboola-tracking-code",
  "cpc": 0.5,
  "daily_cap": 500,
  "spending_limit": 5000,
  "spending_limit_model": "ENTIRE",
  "country_targeting": {
    "type" : "INCLUDE",
    "value" : ["AU", "GB"]
  },
  "sub_country_targeting": null,
  "platform_targeting": {
    "type" : "INCLUDE",
    "value" : ["TBLT", "PHON"]
  },
  "publisher_targeting": null,
  "comments" : null,
  "start_date": "2015-04-24",
  "end_date": "2015-05-24",
  "approval_state" : "APPROVED",
  "is_active": true,
  "spent": 0.0,
  "status": "RUNNING",
  "daily_ad_delivery_model" : "STRICT",
  "publisher_bid_modifier" : {
    "values": [{"target" : "publisher1", "cpc_modification" : 1.5},
               {"target" : "publisher2", "cpc_modification" : 0.9}
    ]
  }
}

```

1.7 Pausing / Unpausing Campaigns

The Campaign's **status** field is *read-only*, and thus cannot be modified via an update request. Instead, one must change the Campaign's **is_active** field. The **is_active** field has two possible values: **true** and **false**. When an Item has both **is_active=true** and **status=RUNNING**, it is considered "active". If the Item has **is_active=false** its **status** will become **PAUSED**.



When a Campaign is **PAUSED**, it is not served in Taboola widgets.

1.7.1 Pause

In order to *pause* a Campaign, send an *update* request in the following form:

```
POST /backstage/api/1.0/[account-id]/campaigns/[campaign-id]/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]
```

```
{
  "is_active" : false
}
```

1.7.1.1 Example Response

200 OK

```
{
  "id": "123",
  "advertiser_id" : "taboola-demo-advertiser",
  "name": "Demo Campaign",
  "branding_text": "Pizza",
  "tracking_code": "",
  "cpc": 0.25,
  "daily_cap": 0.0,
  "spending_limit": 1000,
  "spending_limit_model": "MONTHLY",
  "country_targeting": null,
  "sub_country_targeting": null,
  "platform_targeting": null,
  "publisher_targeting": null,
  "comments" : null,
  "start_date": null,
  "end_date": null,
  "approval_state" : "APPROVED",
  "is_active": false,
  "spent": 0.0,
  "status": "PAUSED",
  "daily_ad_delivery_model" : "ACCELERATED"
}
```

1.7.2 Unpause

In order to *unpause* a Campaign, send an HTTP *POST* request in the following form:

```
POST /backstage/api/1.0/[account-id]/campaigns/[campaign-id]/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]
```

```
{
  "is_active" : true
}
```

1.7.2.1 Example Response

200 OK

```
{
```

```

{id": "123",
"advertiser_id" : "taboola-demo-advertiser",
"name": "Demo Campaign",
"branding_text": "Pizza",
"tracking_code": "",
"cpc": 0.25,
"daily_cap": 0.0,
"spending_limit": 1000,
"spending_limit_model": "MONTHLY",
"country_targeting": null,
"sub_country_targeting": null,
"platform_targeting": null,
"publisher_targeting": null,
"comments" : null,
"start_date": null,
"end_date": null,
"approval_state" : "APPROVED",
"is_active": true,
"spent": 0.0,
"status": "RUNNING",
"daily_ad_delivery_model" : "ACCELERATED"
}

```

1.8 Country / Sub country Campaign targeting

The Campaign's geo targeting can be by country or by sub country. For country targeting possible values are two letter country code. For sub country targeting you may only target sub region / DMA of a single country that is being targeted.



Most updated possible country and sub country codes are accessible via dictionary.



Sub country targeting can only be used if only 1 country is used in country targeting in **INCLUDE** mode, and the region or DMA codes must be relevant to that country. In addition, this can be used for region targeting or DMA targeting (DMA only in US) but not both at the same time.

1.8.1 Example: Updating Campaign's Sub country targeting

```

POST /backstage/api/1.0/taboola-demo-advertiser/campaigns/124
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]

```

```
{
  "country_targeting": {
    "type" : "INCLUDE",
    "value" : ["US"]
  },
  "sub_country_targeting": {
    "type" : "INCLUDE",
    "value" : ["NY"]
  }
}
```

Response:

200 OK

```
{
  "id": "124",
  "advertiser_id" : "taboola-demo-advertiser",
  "name": "Demo Campaign",
  "branding_text": "Yummy Pizza",
  "tracking_code": "taboola-tracking-code",
  "cpc": 0.5,
  "daily_cap": 500,
  "spending_limit": 5000,
  "spending_limit_model": "ENTIRE",
  "country_targeting": {
    "type" : "INCLUDE",
    "value" : ["US"]
  },
  "sub_country_targeting": {
    "type" : "INCLUDE",
    "value" : ["NY"]
  },
  "platform_targeting": {
    "type" : "INCLUDE",
    "value" : ["TBLT", "PHON"]
  },
  "publisher_targeting": null,
  "comments" : null,
  "start_date": "2015-04-24",
  "end_date": "2015-05-24",
  "approval_state" : "APPROVED",
  "is_active": true,
  "spent": 0.0,
  "status": "RUNNING",
  "daily_ad_delivery_model" : "STRICT"
}
```

1.9 Using publisher targeting (blacklisting publishers)

It is possible to prevent your campaign from appearing in specific publishers (please note that this is limited up to 430 publishers per campaign). This is performed via the `publisher_targeting` field of the Campaign. This field can contain an array of "Account IDs" representing publishers to block.

1.9.1 Updating the `publisher_targeting` field

To set up a list of blocked publishers for a certain Campaign, you should send an update request like so:

```
POST /backstage/api/1.0/taboola-demo-advertiser/campaigns/124/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]
```

```
{
  "publisher_targeting": {
    "type": "EXCLUDE",
    "value": ["taboola-demo-publisher"]
  }
}
```

Response:

200 OK

```
{
  "id": "124",
  "advertiser_id": "taboola-demo-advertiser",
  "name": "Demo Campaign - Edited",
  "branding_text": "Pizza",
  "tracking_code": "taboola-track",
  "cpc": 0.25,
  "daily_cap": 100,
  "spending_limit": 1000,
  "spending_limit_model": "MONTHLY",
  "country_targeting": {
    "type": "INCLUDE",
    "value": ["AU", "GB"]
  },
  "platform_targeting": {
    "type": "INCLUDE",
    "value": ["TBLT", "PHON"]
  },
  "publisher_targeting": {
    "type": "EXCLUDE",
    "value": ["taboola-demo-publisher"]
  },
  "comments": null,
  "start_date": "2015-04-30",
  "end_date": "2015-05-30",
  "approval_state": "APPROVED",
  "is_active": true,
  "spent": 0.0,
}
```

```
"status": "RUNNING"
}
```

After performing such an update, this Campaign will stop appearing on this publisher's website.

1.9.2 Fetching a List of Available Publishers

In order to be able to block specific Publishers, one must know the Publisher's `account_id`. To *fetch* a list of all Publisher Accounts in the Taboola network, send an HTTP *GET* request in the following form:

```
GET /backstage/api/1.0/taboola-network/publishers/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]
```



To access this endpoint you must know the Admin Network to which your account belongs. This information is exposed via the Token Details resource.



This endpoint is used for getting a list of all Publishers under an Admin Network.

1.9.2.1 Example Request

```
GET /backstage/api/1.0/taboola-network/publishers/
Host: https://backstage.taboola.com
Authorization: Bearer [access-token]
```

1.9.2.2 Example Response

```
200 OK

{
  "results": [
    {
      "id"       : "124",
      "name"     : "Taboola Demo - Publisher",
      "account_id" : "taboola-demo-publisher"
    },
    // ... more results
  ]
}
```