

# Package: tubern (via r-universe)

October 31, 2024

**Title** R Client for the YouTube Analytics and Reporting API

**Version** 0.2.0

**Maintainer** Gaurav Sood <gsood07@gmail.com>

**Description** Get statistics and reports from YouTube. To learn more about the YouTube Analytics and Reporting API, see <<https://developers.google.com/youtube/reporting/>>.

**URL** <http://github.com/soodoku/tubern>

**BugReports** <http://github.com/soodoku/tubern/issues>

**Depends** R (>= 3.2.0)

**License** MIT + file LICENSE

**LazyData** true

**Imports** httr, jsonlite

**VignetteBuilder** knitr

**Suggests** knitr (>= 1.11), testthat, rmarkdown, xml2, lintr

**RoxygenNote** 6.0.1

**Repository** <https://gojiplus.r-universe.dev>

**RemoteUrl** <https://github.com/gojiplus/tubern>

**RemoteRef** HEAD

**RemoteSha** d813f3fd554755b5e21cf8493852f589c7b5f8c2

## Contents

add_groups . . . . .	2
add_group_item . . . . .	3
delete_group . . . . .	3
delete_group_item . . . . .	4
get_report . . . . .	5
list_groups . . . . .	6
list_group_items . . . . .	7
tubern . . . . .	7

tubern_DELETE . . . . .	8
tubern_GET . . . . .	8
tubern_POST . . . . .	9
tubern_PUT . . . . .	9
update_group . . . . .	10
yt_check_token . . . . .	10
yt_oauth . . . . .	11
<b>Index</b>	<b>12</b>

---

add_groups	<i>Add Groups</i>
------------	-------------------

---

## Description

Add Groups

## Usage

```
add_groups(resource_details, ...)
```

## Arguments

resource\_details  
Named nested list. Required. Must provide: etag, snippet title and contentDetails itemType

...  
Additional arguments passed to `tubern_POST`.

## Value

named list

## References

<https://developers.google.com/youtube/analytics/v1/reference/groups/insert>

## Examples

```
## Not run:
add_groups(list(etag="vponEBg8hrR1yBUX0Hz66Uc5WMk/vyGp6PvFo4RvsFtPoIWeCReyIC8",
ContentDetails = list(itemType="youtube#channel"), snippet = list(title = "hello")))

## End(Not run)
```

---

add_group_item	<i>Add Group Item</i>
----------------	-----------------------

---

**Description**

Add Group Item

**Usage**

```
add_group_item(resource_details, ...)
```

**Arguments**

resource_details	Named nested list. Required. Must provide: groupId, resource.id
...	Additional arguments passed to <a href="#">tubern_POST</a> .

**Value**

named list

**References**

<https://developers.google.com/youtube/analytics/v1/reference/groupItems/insert>

**Examples**

```
## Not run:  
add_group_item(list(groupId = "", resource.id = "hello"))  
  
## End(Not run)
```

---

delete_group	<i>Delete Group</i>
--------------	---------------------

---

**Description**

Delete Group

**Usage**

```
delete_group(id, ...)
```

**Arguments**

id String. Required. “The id parameter specifies the YouTube group ID of the group that is being deleted.”

... Additional arguments passed to `tubern_PUT`.

**Value**

named list

**References**

<https://developers.google.com/youtube/analytics/v1/reference/groups/delete>

**Examples**

```
## Not run:
delete_group(id="ABZZzGSIAAA")

## End(Not run)
```

---

delete\_group\_item      *Delete Group*

---

**Description**

Delete Group

**Usage**

```
delete_group_item(id, ...)
```

**Arguments**

id String. Required. “The id parameter specifies the YouTube group item ID for the group that is being deleted.”

... Additional arguments passed to `tubern_PUT`.

**Value**

named list

**References**

<https://developers.google.com/youtube/analytics/v1/reference/groupItems/delete>

**Examples**

```
## Not run:
delete_group_item(id="ABZZzGSIAAA")

## End(Not run)
```

---

get\_report

*Get Reports*


---

**Description**

Lists reporting jobs that have been scheduled for a channel or content owner.

**Usage**

```
get_report(ids, metrics, start_date = NULL, end_date = NULL,
  currency = NULL, dimensions, filters, historical_channel_data = NULL,
  max_results = NULL, sort = NULL, start_index = NULL, user_ip = NULL,
  ...)
```

**Arguments**

ids	Named vector with two potential names: channel or contentOwner If channel, potential values are: mine or channel_id If contentOwner, potential values are: owner_id of the content
metrics	String. Comma-separated list of YouTube Analytics metrics, such as views or likes, dislikes.
start_date	String. Must be in YYYY-MM-DD format.
end_date	String. Must be in YYYY-MM-DD format.
currency	Optional. String. Default is USD. Specifies what earnings metrics like earnings, adEarnings, grossRevenue, playbackBasedCpm, impressionBasedCpm will be reported in.
dimensions	String. Optional. Comma-separated list of YouTube Analytics dimensions, such as video or ageGroup, gender.
filters	Named Vector. Optional. For instance, "video==pd1FJh59zxQ,Zhawgd0REhA;country==IT restricts the result set to include data for the given videos in Italy"
historical_channel_data	Boolean. Defaults is False. "Whether the API response should include channels' watch time and view data from the time period prior to when the channels were linked to the content owner."
max_results	Integer. Optional. The maximum number of rows to include in the response.
sort	String. Optional A comma-separated list of dimensions or metrics that determine the sort order for YouTube
start_index	Integer. Optional. "The 1-based index of the first entity to retrieve."
user_ip	"IP address of the end user for whom the API call is being made."
...	Additional arguments passed to <a href="#">tubern_GET</a> .

**Value**

named list

**References**

<https://developers.google.com/youtube/analytics/v1/reference/reports/query>

**Examples**

```
## Not run:
get_report(ids = "channel==MINE", metrics = "views",
start_date = "2010-04-01", end_date = "2017-01-01")

## End(Not run)
```

---

list_groups	<i>List Groups</i>
-------------	--------------------

---

**Description**

List Groups

**Usage**

```
list_groups(filter, page_token, ...)
```

**Arguments**

filter	Named Vector. Required. Only One of the two: id or mine. id “The Named parameter specifies a comma-separated list of the YouTube group ID(s) for the resource(s) that are being retrieved.” mine String. Can be 'True' or 'False.' “Set this parameter’s value to true to retrieve all groups owned by the authenticated user.”
page_token	String. Optional. “Identifies a specific page in the result set that should be returned.”
...	Additional arguments passed to <a href="#">tubern_GET</a> .

**Value**

named list

**References**

<https://developers.google.com/youtube/analytics/v1/reference/groups/list>

**Examples**

```
## Not run:  
list_groups(filter = c(mine = 'True'))  
  
## End(Not run)
```

---

list_group_items	<i>List Group Items</i>
------------------	-------------------------

---

**Description**

List Group Items

**Usage**

```
list_group_items(group_id, ...)
```

**Arguments**

group_id	String. Required. ID of the group
...	Additional arguments passed to <a href="#">tubern_GET</a> .

**Value**

named list

**References**

<https://developers.google.com/youtube/analytics/v1/reference/groupItems/list>

**Examples**

```
## Not run:  
list_group_items(group_id = "vponEBg8hrR1yBUX0Hz66Uc5WMk/vyGp6PvFo4RvsFtPoIWeCReyIC8")  
  
## End(Not run)
```

---

tubern	<b>tubern</b> provides access to the YouTube Analytics and Reporting API
--------	--

---

**Description**

**tubern** provides access to the YouTube Analytics and Reporting API

---

tubern_DELETE	<i>DELETE</i>
---------------	---------------

---

**Description**

DELETE

**Usage**

tubern\_DELETE(path, query = NULL, body = "", ...)

**Arguments**

path	path to specific API request URL
query	query list
body	passing image through body
...	Additional arguments passed to <a href="#">DELETE</a> .

**Value**

list

---

tubern_GET	<i>Base POST AND GET functions. Not exported.</i>
------------	---

---

**Description**

GET

**Usage**

tubern\_GET(path, query = NULL, ...)

**Arguments**

path	path to specific API request URL
query	query list
...	Additional arguments passed to <a href="#">GET</a> .

**Value**

list



---

tubern_POST	<i>POST</i>
-------------	-------------

---

**Description**

POST

**Usage**

```
tubern_POST(path, query = NULL, body = "", ...)
```

**Arguments**

path	path to specific API request URL
query	query list
body	passing image through body
...	Additional arguments passed to <a href="#">POST</a> .

**Value**

list

---

tubern_PUT	<i>PUT</i>
------------	------------

---

**Description**

PUT

**Usage**

```
tubern_PUT(path, query = NULL, body = "", ...)
```

**Arguments**

path	path to specific API request URL
query	query list
body	passing image through body
...	Additional arguments passed to <a href="#">PUT</a> .

**Value**

list

---

update_group	<i>Update Groups</i>
--------------	----------------------

---

**Description**

Update Groups

**Usage**

```
update_group(resource_details, ...)
```

**Arguments**

resource_details	Named nested list. Required. Must provide: id, snippet title
...	Additional arguments passed to <a href="#">tubern_PUT</a> .

**Value**

named list

**References**

<https://developers.google.com/youtube/analytics/v1/reference/groups/update>

**Examples**

```
## Not run:  
update_group(list(id="ABZZzGSIAAA", snippet = list(title = "hello")))  
  
## End(Not run)
```

---

yt_check_token	<i>Check if authentication token is in options</i>
----------------	--

---

**Description**

Check if authentication token is in options

**Usage**

```
yt_check_token()
```

yt\_oauth

*Set up Authorization***Description**

The function looks for `.httr-oauth` in the working directory. If it doesn't find it, it expects an application ID and a secret. If you want to remove the existing `.httr-oauth`, set `remove_old_oauth` to `TRUE`. By default, it is set to `FALSE`. The function launches a browser to allow you to authorize the application

**Usage**

```
yt_oauth(app_id = NULL, app_secret = NULL, scope = "analytics",
         token = ".httr-oauth", ...)
```

**Arguments**

<code>app_id</code>	client id; required; no default
<code>app_secret</code>	client secret; required; no default
<code>scope</code>	Character. <code>analytics</code> or <code>monetary-analytics</code> . Required. Default is <code>monetary-analytics</code> . The scopes are largely exchangeable but <code>monetary-analytics</code> yields extra authorizations that come in handy.
<code>token</code>	path to file containing the token. If a path is given, the function will first try to read from it. Default is <code>.httr-oauth</code> in the local directory. So if there is such a file, the function will first try to read from it.
<code>...</code>	Additional arguments passed to <code>oauth2.0_token</code>

**Value**

sets the `google_token` option and also saves `.httr_auth` in the working directory (find out the working directory via `getwd()`)

**References**

<https://developers.google.com/youtube/analytics/v1/reference/>  
<https://developers.google.com/youtube/analytics/v1/reference/> for different scopes

**Examples**

```
## Not run:
yt_oauth("998136489867-5t3tq1g7hbovoj46dreqd6k5kd35ctjn.apps.googleusercontent.com",
        "Mb0St6cQhhFkwETXKur-L9rN")

## End(Not run)
```

# Index

add\_group\_item, [3](#)  
add\_groups, [2](#)

DELETE, [8](#)  
delete\_group, [3](#)  
delete\_group\_item, [4](#)

GET, [8](#)  
get\_report, [5](#)

list\_group\_items, [7](#)  
list\_groups, [6](#)

oauth2.0\_token, [11](#)

POST, [9](#)  
PUT, [9](#)

tubern, [7](#)  
tubern-package (tubern), [7](#)  
tubern\_DELETE, [8](#)  
tubern\_GET, [5-7](#), [8](#)  
tubern\_POST, [2](#), [3](#), [9](#)  
tubern\_PUT, [4](#), [9](#), [10](#)

update\_group, [10](#)

yt\_check\_token, [10](#)  
yt\_oauth, [11](#)