Package ‘brandwatchR’

August 13, 2018

Type Package
Title 'Brandwatch' API to R
Version 0.3.0
URL https://github.com/Phippsy/brandwatchR
BugReports https://github.com/Phippsy/brandwatchR/issues
Description Interact with the 'Brandwatch' API <https://developers.brandwatch.com/docs>. Allows you to authenticate to the API and obtain data for projects, queries, query groups tags and categories. Also allows you to directly obtain mentions and aggregate data for a specified query or query group.
License MIT + file LICENSE
Encoding UTF-8
LazyData true
RoxygenNote 6.1.0
Depends R (>= 3.4.0)
Imports httr (>= 1.3.1), jsonlite (>= 1.5), data.table (>= 1.10)
Suggests testthat
NeedsCompilation no
Author Donal Phipps [aut, cre]
Maintainer Donal Phipps <donal.phipps@gmail.com>
Repository CRAN
Date/Publication 2018-08-13 14:40:16 UTC

R topics documented:

brandwatchR ........................................................................... 2
bwr_auth ............................................................................... 3
bwr_cat_create ................................................................. 4
bwr_cat_get ................................................................. 5
brandwatchR

brandwatchR: A package for retrieving data from the Brandwatch API.

Description

The brandwatchR package provides families of wrapper functions for the Brandwatch API.

Authentication

- `bwr_auth` allows the user to authenticate and cache authentication information for future API calls.

Projects

- `bwr_projects_get` - get a dataframe of projects

Queries / mentions

- `bwr_query_get` - get a dataframe of queries for a project
- `bwr_query_check` - check the syntax of a query
- `bwr_query_create` - upload a new query
- `bwr_query_delete` - delete a specified query
### bwr_auth

Authenticate with the Brandwatch API

**Description**

Authenticates using specified username and password. Also caches a local version of the authentication token unless otherwise specified.

**Usage**

```python
bwr_auth(un = NULL, pw = NULL, refresh = FALSE, cache = TRUE)
```
Arguments

un  Your Brandwatch Username
pw  Your Brandwatch Password
refresh  (Boolean) Should the token be refreshed? If no, a token will be read from a locally cached file `.bw_token` if this file is available. If no token is available, or the refresh argument is TRUE, a new token will be requested and cached from the API.
cache  (Boolean) Can the token be locally cached? If TRUE, the token will be cached in the current working directory in `.bw_token`. If FALSE, the token will be not be cached locally.

Value

Invisibly returns the string containing the token returned by Brandwatch. Your environment variable 'BW_TOKEN' will be set to the value of your authentication token.

Examples

```r
## Not run: bwr_auth(un = 'mickey@mouse.com', pw = 'itsasmallworld')
```

---

**bwr_cat_create**

Upload a new category to Brandwatch

Description

Refer to https://developers.brandwatch.com/docs/creating-categories for more information. Unless otherwise specified, provide a single string argument.

Usage

```r
bwr_cat_create(project_id = NULL, name = NULL, children = NULL,
    token = Sys.getenv("BW_TOKEN"))
```

Arguments

- **project_id**: The project ID into which you’d like to insert a category
- **name**: The name of the category
- **children**: A list of lists (sorry!), each list defining the key-value pairs for name (mandatory) and id (optional) of each child subcategory.
- **token**: The authentication token, obtained using bwr_auth()

Value

Returns a list containing the JSON response from the server.
**bwr_cat_get**

Get a data frame of available categories for a given project

**Description**

See the Brandwatch documentation for more information at https://developers.brandwatch.com/docs/retrieving-categories.

**Usage**

```r
bwr_cat_get(project_id = NULL, token = Sys.getenv("BW_TOKEN"))
```

**Arguments**

- **project_id**  
  The project id you’d like to return all available categories for. Obtain a data frame of projects using `bwr_get_projects()`
- **token**  
  The auth token for this user, which should be generated into an environment variable using `bwr_auth()`

**Value**

Returns a dataframe of categories matching the project.

**Examples**

```r
## Not run: my_categories <- bwr_cat_get(project_id = 12345678)
```

---

**bwr_filters_get**

Get a data frame of available filters for the Brandwatch API

**Description**

See the Brandwatch documentation for more information at https://developers.brandwatch.com/docs/available-filters

**Usage**

```r
bwr_filters_get(token = Sys.getenv("BW_TOKEN"))
```
Arguments

- **token**: The auth token for this user, which should be generated into an environment variable using `bwr_auth()`

Value

Returns a dataframe of available filters

Examples

```r
## Not run: my_queries <- bwr_query_get(project_id = 12345678)
```

**bwr_mentions_chart**

*Get a dataframe of chart-ready data for a specified Brandwatch query or query group*

Description

For more information, please refer to https://developers.brandwatch.com/docs/chart-dimensions-and-aggregates

Usage

```r
bwr_mentions_chart(project_id = NULL, query_id = NULL, querygrp_id = NULL, date_range = c(Sys.Date() - 31, Sys.Date() - 1), aggregate = NULL, dimension1 = NULL, dimension2 = NULL, filters = NULL, token = Sys.getenv("BW_TOKEN"))
```

Arguments

- **project_id**: The project id in which the specified query is contained. Obtain a list of project IDs using `bwr_get_projects()`.
- **query_id**: The query ID you’d like to see tweeter data for. Note that you can only specify a query or a query group, not both.
- **querygrp_id**: The query group ID you’d like to see total mentions for. Note that you can only specify a query or a query group, not both.
- **date_range**: A character vector containing 2 date values in YYYY-mm-dd format. The first value is the beginning of your desired date range and the second value is the end of the date range.
- **aggregate**: The numeric variable which you’d like to return results for (e.g. ‘volume’)
- **dimension1**: The first dimension which you’d like to split your data by (e.g. ‘months’) (Optional)
- **dimension2**: The second dimension which you’d like to split your data by (e.g. ‘countries’) (Optional)
- **filters**: A list of key-value pairs to filter the mentions query by. Use the `bwr_filters_get()` function to find out all available filters.
- **token**: The authentication token, acquired using `bwr_auth()`
Value

Returns a data frame (including some list-columns) of the JSON results, aggregated according to your 2 dimensions.

Examples

```r
## not run: my_project <- bwr_projects_get()[1]
my_query <- bwr_query_get(project_id = my_project)$id[1]
my_mentions_chart <- bwr_mentions_chart(project_id = my_project,
query_id = my_query,
date_range = c('2018-01-01', '2018-02-01'),
aggregate = "month",
dimension1 = "sentiment",
dimension2 = "volume")

## End(not run)
```

**bwr_mentions_get**

*Get a list of mentions for the specified query and project*

**Description**

Returns a data frame containing any results found for the specified query ID.

**Usage**

```r
bwr_mentions_get(project_id = NULL, query_id = NULL,
date_range = c(Sys.Date() - 31, Sys.Date() - 1), order_by = NULL,
order_direction = NULL, filters = NULL, page = NULL,
token = Sys.getenv("BW_TOKEN"))
```

**Arguments**

- `project_id`: Numeric. The project id in which the specified query is contained. Obtain a list of project IDs using `bwr_get_projects()`.
- `query_id`: Numeric. The id of the query you’d like to run. Obtain a list of queries for any specified project using `bwr_get_queries()`.
- `date_range`: A character vector containing 2 date values in YYYY-mm-dd format. The first value is the beginning of your desired date range and the second value is the end of the date range.
- `order_by`: String. (Optional) The name of the metric you’d like to order your results by.
- `order_direction`: String. Either ‘asc’ or ‘desc’, to specify ascending or descending order.
- `filters`: (Optional) A list of key-value pairs to filter the mentions query by. Use the `bwr_filters_get()` function to find out all available filters.
bwr_mentions_topauthors

Get a dataframe of top author data for a specified Brandwatch query or query group

Description

Get a dataframe of top author data for a specified Brandwatch query or query group

Usage

```r
bwr_mentions_topauthors(project_id = NULL, query_id = NULL,
querygrp_id = NULL, date_range = c(Sys.Date() - 31, Sys.Date() - 1),
filters = NULL, token = Sys.getenv("BW_TOKEN"))
```
Arguments

project_id     The project id in which the specified query is contained. Obtain a list of project IDs using bwr_get_projects().

query_id      The query ID you'd like to see author data for. Note that you can only specify a query or a query group, not both.

querygrp_id   The query group ID you'd like to see total mentions for. Note that you can only specify a query or a query group, not both.

date_range    A character vector containing 2 date values in YYYY-mm-dd format. The first value is the beginning of your desired date range and the second value is the end of the date range.

filters       (Optional) A list of key-value pairs to filter the mentions query by. Use the bwr_filters_get() function to find out all available filters.

token         The authentication token, acquired using bwr_auth()

Value

Returns a data frame (including some list-columns) of the JSON results, showing information about the top authors related to your query.

Examples

## Not run: my_project <- bwr_projects_get(IDid[1]
my_query <- bwr_query_get(project_id = my_project)$id[1]
my_mentions <- bwr_mentions_topauthors(project_id = my_project,
  query_id = my_query,
  date_range = c('2018-01-01', '2018-02-01'))
## End(Not run)
Arguments

project_id  The project id in which the specified query is contained. Obtain a list of project IDs using \texttt{bwr\_get\_projects()}.  

query_id  The query ID you’d like to see total mentions for. Note that you can only specify a query or a query group, not both.  

querygrp_id  The query group ID you’d like to see total mentions for. Note that you can only specify a query or a query group, not both.  

date_range  A character vector containing 2 date values in YYYY-mm-dd format. The first value is the beginning of your desired date range and the second value is the end of the date range.  

order_by  Metric by which you wish to order the topics returned. Can be either ‘volume’ (the number of Mentions the topic appears in) or ‘burst’ (the rate that the topic has emerged over time).  

limit  The maximum number of topics to be returned.  

exclude_categories  You have the option to exclude Categories from the topics returned (this is often done in the Brandwatch Analytics UI).  

exclude_tags  The authentication token, acquired using \texttt{bwr\_auth()}.  

filters  (Optional) A list of key-value pairs to filter the mentions query by. Use the \texttt{bwr\_filters\_get()} function to find out all available filters.  

token  The authentication token, acquired using \texttt{bwr\_auth()}.

Value

Returns a data frame (including some list-columns) of the JSON results, showing information about topics related to your query.

Examples

```r
## not run: my_project <- bwr_projects_get(IDid[Q]
my_query <- bwr_query_get(project_id = my_project)$id[1]
my_mentions <- bwr_mentions_topics(project_id = my_project,
query_id = my_query,
date_range = c('2018-01-01', '2018-02-01'))
## end(not run)
```

\textbf{bwr\_mentions\_topsites} \hspace{1cm} \textit{Get a dataframe of top site data for a specified Brandwatch query or query group}

Description

Get a dataframe of top site data for a specified Brandwatch query or query group.
bwr_mentions_toptweeters

Usage

```r
bwr_mentions_topsites(project_id = NULL, query_id = NULL,
                        querygrp_id = NULL, date_range = c(Sys.Date() - 31, Sys.Date() - 1),
                        filters = NULL, token = Sys.getenv("BW_TOKEN"))
```

Arguments

- **project_id**: The project id in which the specified query is contained. Obtain a list of project IDs using `bwr_get_projects()`.
- **query_id**: The query ID you’d like to see total mentions for. Note that you can only specify a query or a query group, not both.
- **querygrp_id**: The query group ID you’d like to see total mentions for. Note that you can only specify a query or a query group, not both.
- **date_range**: A character vector containing 2 date values in YYYY-mm-dd format. The first value is the beginning of your desired date range and the second value is the end of the date range.
- **filters**: (Optional) A list of key-value pairs to filter the mentions query by. Use the `bwr_filters_get()` function to find out all available filters.
- **token**: The authentication token, acquired using `bwr_auth()`

Value

Returns a data frame (including some list-columns) of the JSON results, showing information about the top sites related to your query.

Examples

```r
## Not run: my_project <- bwr_projects_get()$id[1]
my_query <- bwr_query_get(project_id = my_project)$id[1]
my_mentions <- bwr_mentions_topsites(project_id = my_project,
                                      query_id = my_query,
                                      date_range = c('2018-01-01', '2018-02-01'))

## End(Not run)
```
Usage

bwr_mentions_toptweeters(project_id = NULL, query_id = NULL, querygrp_id = NULL, date_range = c(Sys.Date() - 31, Sys.Date() - 1), filters = NULL, token = Sys.getenv("BW_TOKEN"))

Arguments

project_id  The project id in which the specified query is contained. Obtain a list of project IDs using bwr_get_projects().
query_id  The query ID you’d like to see tweeter data for. Note that you can only specify a query or a query group, not both.
querygrp_id  The query group ID you’d like to see total mentions for. Note that you can only specify a query or a query group, not both.
date_range  A character vector containing 2 date values in YYYY-mm-dd format. The first value is the beginning of your desired date range and the second value is the end of the date range.
filters  (Optional) A list of key-value pairs to filter the mentions query by. Use the bwr_filters_get() function to find out all available filters.
token  The authentication token, acquired using bwr_auth()

Value

Returns a data frame (including some list-columns) of the JSON results, showing information about the top tweeters related to your query.

Examples

## not run: my_project <- bwr_projects_get()$id[1]
my_query <- bwr_query_get(project_id = my_project)$id[1]
my_mentions <- bwr_mentions_toptweeters(project_id = my_project,
query_id = my_query,
date_range = c('2018-01-01', '2018-02-01'))
## End(not run)
**Arguments**

- **project_id**  
The project id in which the specified query is contained. Obtain a list of project IDs using `bwr_get_projects()`.

- **query_id**  
The query ID you’d like to see total mentions for. Note that you can only specify a query or a query group, not both.

- **querygrp_id**  
The query group ID you’d like to see total mentions for. Note that you can only specify a query or a query group, not both.

- **filters**  
(Optional) A list of key-value pairs to filter the mentions query by. Use the `bwr_filters_get()` function to find out all available filters.

- **date_range**  
A character vector containing 2 date values in YYYY-mm-dd format. The first value is the beginning of your desired date range and the second value is the end of the date range.

- **token**  
The authentication token, acquired using `bwr_auth()`

**Value**

Returns a numeric vector of length 1 with the results.

**Examples**

```r
## not run: my_project <- bwr_projects_get()$id[1]
my_query <- bwr_queries_get(project_id = my_project)$id[1]
my_mentions <- bwr_mentions_total(
  project_id = my_project,
  query_id = my_query,
  date_range = c('2018-01-01', '2018-02-01'))
## end(not run)
```

**bwr_metrics_get**  
*Get a data frame of possible metric values for the Brandwatch API*

**Description**

When examining parameters which can be used to filter a metric call, some parameters have a limited set of acceptable values. You can find out the current acceptable values using this function.

**Usage**

```r
bwr_metrics_get(token = Sys.getenv("BW_TOKEN"))
```

**Arguments**

- **token**  
The auth token for this user, which should be generated into an environment variable using `bwr_auth()`
bwr_projects_get

Details

See the Brandwatch documentation for more information at https://developers.brandwatch.com/docs/available-filters

Value

Returns a dataframe of available filters

Examples

```r
## not run: my_queries <- bwr_metrics_get(project_id = 1234534)
```

---

---

**bwr_projects_get**

*Get Brandwatch project data*

---

**Description**

#' Read the Brandwatch documentation for this API call at https://developers.brandwatch.com/docs/retrieving-projects

**Usage**

```
bwr_projects_get(target = NULL, token = Sys.getenv("BW_TOKEN"))
```

**Arguments**

- **target** *(Optional)* the target for your query. If not specified, results will include all available projects for the authenticated user and all available data for each project. If 'summary', results will be a more simplified list of projects and excludes richer data such as query links.

- **token** The Brandwatch token, which will be automatically available via environment arguments provided you run bwr_auth() prior to any other functions.

**Value**

Returns a data frame of project data.

**Examples**

```r
## not run: my_projects <- bwr_get_projects(target = 'summary')
```
**bwr_querygrp_delete**  
*Delete a specified Brandwatch query group*

**Description**
Delete a specified Brandwatch query group

**Usage**
```r
bwr_querygrp_delete(project_id = NULL, group_id = NULL,  
token = Sys.getenv("BW_TOKEN"))
```

**Arguments**
- `project_id`: The project ID in which your target query group can be found.
- `group_id`: The query group ID which you’d like to delete. Find out the list of query groups in a given project using `bwr_querygrp_get()`. 
- `token`: The API authentication token, obtained using `bwr_auth()`

**Value**
Returns a list of the JSON response from the server.

**Examples**
```r
## Not run: kill <- bwr_querygrp_delete(project_id = 234234234, group_id = 23423423)
```

---

**bwr_querygrp_get**  
*Get a data frame of Brandwatch query groups in the specified project*

**Description**
Get a data frame of Brandwatch query groups in the specified project

**Usage**
```r
bwr_querygrp_get(project_id = NULL, token = Sys.getenv("BW_TOKEN"))
```

**Arguments**
- `project_id`: The project ID whose query groups you’d like to return
- `token`: The authentication token, obtained using `bwr_auth()`

**Value**
Returns a data frame of the query groups for the given project, if any exist.
bwr_query_check

Validate a Brandwatch query

Description

Validate a Brandwatch query

Usage

bwr_query_check(query = list(), token = Sys.getenv("BW_TOKEN"))

Arguments

query A list containing named parameters: query and language.
token The authentication token, acquired using bwr_auth()

Value

Returns a list containing any errors or issues, if found

Examples

## not run: any_issues <- bwr_query_check(query = list(query = 'at_mentions:huey', language = 'en'))

bwr_query_create

Upload a new query to Brandwatch

Description

Refer to https://developers.brandwatch.com/docs/creating-queries for more information. Unless otherwise specified, provide a single string argument.

Usage

bwr_query_create(project_id = NULL, type = "search string", languageAgnostic = FALSE, samplePercent = 100, languages = "en", includedTerms = NULL, name = NULL, description = "My API query", industry = "general", token = Sys.getenv("BW_TOKEN"))
bwr_query_delete

Arguments

- project_id: The project ID into which you’d like to insert a query
- type: The type of query (e.g. 'search string')
- languageAgnostic: (Boolean) Is the query language agnostic?
- samplePercent: (Numeric) The sample percent for the given query
- languages: Which languages to include
- includedTerms: The query syntax to be included.
- name: The name of the query
- description: The description of the query
- industry: The industry classification
- token: The authentication token, obtained using bwr_auth()

Value

Returns a list containing the JSON response from the server.

Examples

```r
## Not run: bwr_query_create(project_id = 12423432,
samplePercent = 50,
includedTerms = 'at_mentions:mickeymouse',
name = 'Sample API query')

## End(Not run)
```

---

bwr_query_delete Delete a specified Brandwatch query

Description

Delete a specified Brandwatch query

Usage

```r
bwr_query_delete(project_id, query_id, token = Sys.getenv("BW_TOKEN"))
```

Arguments

- project_id: The project ID in which the target query can be found. Obtain a data frame of project IDs using bwr_projects_get().
- query_id: The ID of the query which you’d like to delete. Obtain a list of query IDs using bwr_query_get().
- token: The authentication token, obtained using bwr_auth()
bwr_query_get

Value

Returns a list of the JSON response.

Examples

## not run: bwr_query_delete(project_id = 122445, query_id = 23432424)

```
## not run: my_queries <- bwr_query_get(project_id = 12334534)
```

---

### bwr_query_get

**Get a data frame of available queries for a given project**

**Description**

See the Brandwatch documentation for more information at https://developers.brandwatch.com/docs/retrieving-queries.

**Usage**

```
bwr_query_get(project_id = NULL, type = NULL,
               token = Sys.getenv("BW_TOKEN"))
```

**Arguments**

- **project_id**: The project id you’d like to return all available queries for. Obtain a data frame of projects using `bwr_get_projects()`.
- **type** *(Optional)*: The type of query you’d like to return. If not specified, the API will return all available queries for the given project. Currently, results don’t seem to be affected by the type parameter - the API appears to return full results in all cases.
- **token**: The auth token for this user, which should be generated into an environment variable using `bwr_auth()`.

**Value**

Returns a dataframe of queries matching the project.

**Examples**

## not run: my_queries <- bwr_query_get(project_id = 12334534)
**bwr_rule_delete**

Delete a specified Brandwatch rule

**Description**

Delete a specified Brandwatch rule

**Usage**

```r
bwr_rule_delete(project_id, rule_id, token = Sys.getenv("BW_TOKEN"))
```

**Arguments**

- **project_id**
  
  The project ID in which the target rule can be found. Obtain a data frame of project IDs using `bwr_projects_get()`.

- **rule_id**
  
  The ID of the rule which you’d like to delete. Obtain a list of rule IDs using `bwr_rule_get()`.

- **token**
  
  The authentication token, obtained using `bwr_auth()`

**Value**

Returns a list of the JSON response.

**Examples**

```r
## Not run: bwr_rule_delete(project_id = 122445, rule_id = 23432424)
```

---

**bwr_rule_get**

Get a data frame of available rules for a given project

**Description**

See the Brandwatch documentation for more information at https://developers.brandwatch.com/docs/retrieving-rules.

**Usage**

```r
bwr_rule_get(project_id = NULL, token = Sys.getenv("BW_TOKEN"))
```

**Arguments**

- **project_id**
  
  The project id you’d like to return all available rules for. Obtain a data frame of projects using `bwr_get_projects()`

- **token**
  
  The auth token for this user, which should be generated into an environment variable using `bwr_auth()`
Value
Returns a dataframe of rules matching the project. Due to the structure of the API response, some of the dataframe columns are list-columns (i.e. nested dataframes).

Examples
## not run: my_rules <- bwr_rules_get(project_id = 1234534)

---

**bwr_tag_create**

Upload a new tag to Brandwatch

Description
Refer to https://developers.brandwatch.com/docs/creating-tags for more information. Unless otherwise specified, provide a single string argument.

Usage

```r
bwr_tag_create(project_id = NULL, name = NULL, 
    token = Sys.getenv("BW_TOKEN"))
```

Arguments

- **project_id**: The project ID into which you’d like to insert a tag
- **name**: The name of the tag
- **token**: The authentication token, obtained using bwr_auth()

Value
Returns a list containing the JSON response from the server.

Examples
## not run: bwr_tag_create(project_id = 12423432, name = 'Quick test tag')
**bwr_tag_delete**  
*Delete a specified Brandwatch tag*

**Description**
Delete a specified Brandwatch tag

**Usage**
```r
bwr_tag_delete(project_id, tag_id, token = Sys.getenv("BW_TOKEN"))
```

**Arguments**
- `project_id`: The project ID in which the target tag can be found. Obtain a data frame of project IDs using `bwr_projects_get()`.
- `tag_id`: The ID of the tag which you’d like to delete. Obtain a list of tag IDs using `bwr_tag_get()`.
- `token`: The authentication token, obtained using `bwr_auth()`

**Value**
Returns a list of the JSON response.

**Examples**
```r
## Not run: bwr_tag_delete(project_id = 122445, tag_id = 23432424)
```

---

**bwr_tag_get**  
*Get a data frame of available tags for a given project*

**Description**
See the Brandwatch documentation for more information at https://developers.brandwatch.com/docs/retrieving-tags.

**Usage**
```r
bwr_tag_get(project_id = NULL, token = Sys.getenv("BW_TOKEN"))
```

**Arguments**
- `project_id`: The project id you’d like to return all available tags for. Obtain a data frame of projects using `bwr_get_projects()`.
- `token`: The auth token for this user, which should be generated into an environment variable using `bwr_auth()`
bwr_wf_get

Value

Returns a dataframe of tags matching the project.

Examples

```r
## Not run: my_tags <- bwr_tag_get(project_id = 1234534)
```

bwr_wf_get  
*Get a list of workflows for the specified query and project*

Description

Refer to https://developers.brandwatch.com/docs/retrieving-workflow for more information.

Usage

```r
bwr_wf_get(project_id = NULL, token = Sys.getenv("BW_TOKEN"))
```

Arguments

- `project_id`: The project id in which the specified workflow is contained. Obtain a list of project IDs using `bwr_get_projects()`.
- `token`: The authentication token, acquired using `bwr_auth()`

Value

Returns a list containing all results

Examples

```r
## Not run: my_project <- bwr_get_projects()$id[1]
my_workflow <- bwr_wf_get(project_id = my_project)
## End(Not run)
```
Index

brandwatchR, 2
brandwatchR-package (brandwatchR), 2
bwr_auth, 2, 3, 8
bwr_cat_create, 3, 4
bwr_cat_get, 3, 5
bwr_filters_get, 3, 5, 8
bwr_mentions_chart, 6
bwr_mentions_get, 3, 7
bwr_mentions_topauthors, 8
bwr_mentions_topics, 3, 9
bwr_mentions_topsites, 3, 10
bwr_mentions_toptweeters, 3, 11
bwr_mentions_total, 3, 12
bwr_metrics_get, 3, 13
bwr_projects_get, 2, 8, 14
bwr_query_check, 2, 16
bwr_query_create, 2, 16
bwr_query_delete, 2, 17
bwr_query_get, 2, 8, 18
bwr_querygrp_delete, 3, 15
bwr_querygrp_get, 3, 15
bwr_rule_delete, 19
bwr_rule_get, 19
bwr_tag_create, 3, 20
bwr_tag_delete, 3, 21
bwr_tag_get, 3, 21
bwr_wf_get, 22