

Package ‘rdrop2’

July 31, 2015

Title Programmatic Interface to the 'Dropbox' API

Version 0.7.0

Description Provides full programmatic access to the Dropbox file hosting platform (dropbox.com), including support for all standard file operations.

Depends R (>= 3.1.1)

License MIT + file LICENSE

BugReports <https://github.com/karthik/rdrop2/issues>

LazyData true

Imports assertthat, data.table, dplyr, httr, jsonlite, magrittr

Suggests testthat, uuid

NeedsCompilation no

Author Karthik Ram [aut, cre]

Maintainer Karthik Ram <karthik.ram@gmail.com>

Repository CRAN

Date/Publication 2015-07-31 01:13:23

R topics documented:

drop_acc	2
drop_auth	3
drop_copy	3
drop_create	4
drop_delete	5
drop_delta	6
drop_dir	6
drop_exists	8
drop_get	9
drop_history	10
drop_media	10
drop_move	11
drop_read_csv	12

drop_search	13
drop_share	14
drop_upload	14

Index	16
--------------	-----------

drop_acc	<i>Dropbox account information</i>
----------	------------------------------------

Description

Retrieves information about the user's account. Returns the following fields: referral_link, display_name, uid, locale, email_verified, quota_info.datastores, quota_info.shared, quota_info.quota, quota_info.normal, is_paired, country, name_details.familiar_name, name_details.surname, name_details.given_name, email

Usage

```
drop_acc(dtoken = get_dropbox_token(), verbose = FALSE)
```

Arguments

dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.
verbose	By default verbose output is FALSE. Set to TRUE if you need to troubleshoot any output or grab additional parameters.

Examples

```
## Not run:
drop_acc()
# Select name and UID
drop_acc() %>% select(uid, display_name, quota_info.normal)

## End(Not run)
```

drop_auth

Authentication for Dropbox

Description

This function authenticates you into Dropbox. The documentation for the [core Dropbox API](#) provides more details including alternate methods if you desire to reimplement your own.

Usage

```
drop_auth(new_user = FALSE, key = "mmhfsybfddom42w",  
          secret = "18zeqqgm1ne5z0", cache = TRUE)
```

Arguments

new_user	Default is FALSE. Set to TRUE if you need to switch to a new user account or just flush existing token.
key	Your application key. rdrop2 already comes with a key/secret but you are welcome to swap out with our own. Since these keys are shipped with the package, there is a small chance they could be voided if someone abuses the key. So if you plan to use this in production, or for an internal tool, the recommended practice is to create a new application on Dropbox and use those keys for your purposes.
secret	Your application token. rdrop2 already comes with a key/secret but you are welcome to swap out with our own.
cache	By default your credentials are locally cached in a file called <code>.http-oauth</code> . Set to FALSE if you need to authenticate separately each time.

Examples

```
## Not run:  
drop_auth()  
# If you want to overwrite an existing token and switch to a new user,  
# set new_user to TRUE.  
drop_auth(new_user = TRUE)  
  
## End(Not run)
```

drop_copy

Copies a file or folder to a new location.

Description

Copies a file or folder to a new location.

Usage

```
drop_copy(from_path = NULL, to_path = NULL, root = "auto",
          verbose = FALSE, dtoken = get_dropbox_token())
```

Arguments

from_path	Source file or folder
to_path	destination file or folder
root	This is required. The root relative to which path is specified. Valid values are auto (recommended and also the default), sandbox, and dropbox.
verbose	By default verbose output is FALSE. Set to TRUE if you need to troubleshoot any output or grab additional parameters.
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:
write.csv(mtcars, file = "mt.csv")
drop_upload("mt.csv")
drop_create("drop_test2")
drop_copy("mt.csv", "drop_test2/mt2.csv")

## End(Not run)
```

drop_create	<i>Creates a folder on Dropbox</i>
-------------	------------------------------------

Description

Returns a list containing the following fields: "size", "rev", "thumb_exists", "bytes", "modified", "path", "is_dir", "icon", "root", "revision"

Usage

```
drop_create(path = NULL, root = "auto", verbose = FALSE,
            dtoken = get_dropbox_token())
```

Arguments

path	This is required The path to the new folder to create relative to root.
root	This is required. The root relative to which path is specified. Valid values are auto (recommended and also the default), sandbox, and dropbox.

verbose	By default verbose output is FALSE. Set to TRUE if you need to troubleshoot any output or grab additional parameters.
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:
drop_create(path = "foobar")

## End(Not run)
```

drop_delete	<i>Deletes a file or folder.</i>
-------------	----------------------------------

Description

Deletes a file or folder.

Usage

```
drop_delete(path = NULL, root = "auto", verbose = FALSE,
            dtoken = get_dropbox_token())
```

Arguments

path	This is required The path to the new folder to create relative to root.
root	This is required. The root relative to which path is specified. Valid values are auto (recommended and also the default), sandbox, and dropbox.
verbose	By default verbose output is FALSE. Set to TRUE if you need to troubleshoot any output or grab additional parameters.
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

drop_delta	<i>Get a list of Deltas</i>
------------	-----------------------------

Description

Get a list of Deltas

Usage

```
drop_delta(cursor = NULL, locale = NULL, path_prefix = NULL,
           include_media_info = NULL, dtoken = get_dropbox_token())
```

Arguments

cursor	The last cursor
locale	Dropbox uses the locale parameter to specify language settings of content responses. If your app supports any language other than English, insert the appropriate IETF language tag. When a supported language is specified, Dropbox will returned translated size and/or user_error fields (where applicable)
path_prefix	The path to subset
include_media_info	Set to TRUE
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:
z <- drop_delta(path_prefix = "/Public")
# If no files have changed during this time, entries will be NULL
drop_delta(cursor = z$cursor, path_prefix = "/Public")

## End(Not run)
```

drop_dir	<i>drop_dir</i>
----------	-----------------

Description

Retrieves file and folder metadata.

Usage

```
drop_dir(path = NULL, file_limit = 10000, hash = NULL, list = TRUE,
         include_deleted = FALSE, rev = NULL, locale = NULL, n = 25,
         include_media_info = TRUE, include_membership = FALSE, verbose = FALSE,
         dtoken = get_dropbox_token())
```

Arguments

path	This is required The path to the new folder to create relative to root.
file_limit	Default is 10000. (max is 25000). When listing a folder, the service won't report listings containing more than the specified amount of files and will instead respond with a 406 (Not Acceptable) status response.
hash	Each call to /metadata on a folder will return a hash field, generated by hashing all of the metadata contained in that response. On later calls to /metadata, you should provide that value via this parameter so that if nothing has changed, the response will be a 304 (Not Modified) status code instead of the full, potentially very large, folder listing. This parameter is ignored if the specified path is associated with a file or if list=false.
list	The strings true and false are valid values. true is the default. If true, the folder's metadata will include a contents field with a list of metadata entries for the contents of the folder. If false, the contents field will be omitted.
include_deleted	Only applicable when list is set. If this parameter is set to true, then contents will include the metadata of deleted children. Note that the target of the metadata call is always returned even when it has been deleted (with is_deleted set to true) regardless of this flag.
rev	If you include a particular revision number, then only the metadata for that revision will be returned.
locale	Dropbox uses the locale parameter to specify language settings of content responses. If your app supports any language other than English, insert the appropriate IETF language tag. When a supported language is specified, Dropbox will returned translated size and/or user_error fields (where applicable)
n	Number of entries to print. Default is 25
include_media_info	If true, each file will include a photo_info dictionary for photos and a video_info dictionary for videos with additional media info. If the data isn't available yet, the string pending will be returned instead of a dictionary.
include_membership	If true, metadata for a shared folder will include a list of members and a list of groups.
verbose	By default verbose output is FALSE. Set to TRUE if you need to troubleshoot any output or grab additional parameters.
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Details

A sample output for contents returns the following fields: size, rev, thumb_exists, bytes, modified, client_mtime, path, is_dir, icon, root, mime_type, revision. Note that with verbose set to FALSE you only get a subset of these fields. To get the full output, look in drop_dir()\$contents

Examples

```
## Not run:
drop_dir()

## End(Not run)
```

drop_exists

Checks to see if a file/folder exists on Dropbox

Description

Since many file operations such as move, copy, delete and history can only act on files that currently exist on a Dropbox store, checking to see if the path is valid before operating prevents bad API calls from being sent to the server. This functions returns a logical response after checking if a file path is valid on Dropbox.

Usage

```
drop_exists(path = NULL, dtoken = get_dropbox_token())
```

Arguments

path	The full path to a Dropbox file
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:
drop_create("existential_test")
drop_exists("existential_test")
drop_delete("existential_test")

## End(Not run)
```

drop_get	<i>Downloads a file from Dropbox</i>
----------	--------------------------------------

Description

Downloads a file from Dropbox

Usage

```
drop_get(path = NULL, local_file = NULL, rev = "", overwrite = FALSE,  
         verbose = FALSE, progress = FALSE, dtoken = get_dropbox_token())
```

Arguments

path	This is required The path to the new folder to create relative to root.
local_file	The name of the local copy. Leave this blank if you're fine with the original name.
rev	The revision of the file to retrieve. This defaults to the most recent revision.
overwrite	Default is FALSE but can be set to TRUE.
verbose	By default verbose output is FALSE. Set to TRUE if you need to troubleshoot any output or grab additional parameters.
progress	Progress bars are turned off by default. Set to TRUE ot turn this on. Progress is only reported when file sizes are known. Otherwise just bytes downloaded.
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:  
drop_get(path = 'dataset.zip', local_file = "~/Desktop")  
# To overwrite the existing file  
drop_get(path = 'dataset.zip', overwrite = TRUE)  
  
## End(Not run)
```

drop_history	<i>Obtains metadata for all available revisions of a file, including the current revision.</i>
--------------	--

Description

Only revisions up to thirty days old are available (or more if the Dropbox user has Extended Version History). You can use the revision number in conjunction with the `/restore` call to revert the file to its previous state.

Usage

```
drop_history(path = NULL, dtoken = get_dropbox_token())
```

Arguments

path	This is required The path to the new folder to create relative to root.
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:
write.csv(iris, file = "iris.csv")
drop_upload("iris.csv")
write.csv(iris[iris$Species == "setosa", ], file = "iris.csv")
drop_upload("iris.csv")
drop_history("iris.csv")

## End(Not run)
```

drop_media	<i>Returns a link directly to a file.</i>
------------	---

Description

Similar to `drop_shared`. The difference is that this bypasses the Dropbox webserver, used to provide a preview of the file, so that you can effectively stream the contents of your media. This URL should not be used to display content directly in the browser. **IMPORTANT:** The media link will expire after 4 hours. So you'll need to cache the content with `knitr cache` OR re-run the function call after expiry.

Usage

```
drop_media(path = NULL, locale = NULL, dtoken = get_dropbox_token())
```

Arguments

path	This is required The path to the new folder to create relative to root.
locale	Dropbox uses the locale parameter to specify language settings of content responses. If your app supports any language other than English, insert the appropriate IETF language tag. When a supported language is specified, Dropbox will returned translated size and/or user_error fields (where applicable)
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:
drop_media('public/gifs/duck_rabbit.gif')

## End(Not run)
```

drop_move	<i>Moves a file or folder to a new location.</i>
-----------	--

Description

Moves a file or folder to a new location.

Usage

```
drop_move(from_path = NULL, to_path = NULL, root = "auto",
          verbose = FALSE, dtoken = get_dropbox_token())
```

Arguments

from_path	Source file or folder
to_path	destination file or folder
root	This is required. The root relative to which path is specified. Valid values are auto (recommended and also the default), sandbox, and dropbox.
verbose	By default verbose output is FALSE. Set to TRUE if you need to troubleshoot any output or grab additional parameters.
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:
write.csv(mtcars, file = "mt.csv")
drop_upload("mt.csv")
drop_create("drop_test2")
drop_move("mt.csv", "drop_test2/mt.csv")

## End(Not run)
```

drop_read_csv

drop_read_csv

Description

A lightweight wrapper around `read.csv` to read csv files from Dropbox into memory

Usage

```
drop_read_csv(file, dest = tempdir(), dtoken = get_dropbox_token(), ...)
```

Arguments

<code>file</code>	Name of file with full path relative to Dropbox root
<code>dest</code>	A temporary directory where a csv file is downloaded before being read into memory
<code>dtoken</code>	The Dropbox token generated by drop_auth . <code>rdrop2</code> will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.
<code>...</code>	Additional arguments into <code>read.csv</code>

Examples

```
## Not run:
write.csv(iris, file = "iris.csv")
drop_upload("iris.csv")
# Now let's read this back into an R session
new_iris <- drop_read_csv("iris.csv")

## End(Not run)
```

drop_search	<i>Returns metadata for all files and folders whose filename contains the given search string as a substring.</i>
-------------	---

Description

Returns metadata for all files and folders whose filename contains the given search string as a substring.

Usage

```
drop_search(query = NULL, path = NULL, file_limit = 1000,
            include_deleted = FALSE, locale = NULL, include_membership = FALSE,
            dtoken = get_dropbox_token())
```

Arguments

query	The search string. This string is split (on spaces) into individual words. Files and folders will be returned if they contain all words in the search string.
path	This is required The path to the new folder to create relative to root.
file_limit	Default is 10000. (max is 25000). When listing a folder, the service won't report listings containing more than the specified amount of files and will instead respond with a 406 (Not Acceptable) status response.
include_deleted	If this parameter is set to true, then files and folders that have been deleted will also be included in the search.
locale	Dropbox uses the locale parameter to specify language settings of content responses. If your app supports any language other than English, insert the appropriate IETF language tag. When a supported language is specified, Dropbox will returned translated size and/or user_error fields (where applicable)
include_membership	If true, metadata for a shared folder will include a list of members and a list of groups.
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:
# If you know me, you know why this query exists
drop_search('gif') %>% select(path, is_dir, mime_type)

## End(Not run)
```

drop_share	<i>Creates and returns a shared link to a file or folder.</i>
------------	---

Description

Returns a list with the share url, expires, and visibility.

Usage

```
drop_share(path = NULL, locale = NULL, short_url = TRUE,
           dtoken = get_dropbox_token())
```

Arguments

path	This is required The path to the new folder to create relative to root.
locale	Dropbox uses the locale parameter to specify language settings of content responses. If your app supports any language other than English, insert the appropriate IETF language tag. When a supported language is specified, Dropbox will returned translated size and/or user_error fields (where applicable)
short_url	By default the function return shortened URL. Set to FALSE if you require a full path.
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:
write.csv(mtcars, file = "mt.csv")
drop_upload("mt.csv")
drop_share("mt.csv")

## End(Not run)
```

drop_upload	<i>Uploads a file to Dropbox using PUT semantics.</i>
-------------	---

Description

This function will allow you to write files of any size to Dropbox(even ones that cannot be read into memory) by uploading them in chunks.

Usage

```
drop_upload(file, dest = NULL, overwrite = TRUE, autorename = FALSE,  
            verbose = FALSE, encode = "multipart", dtoken = get_dropbox_token())
```

Arguments

file	Relative path to local file.
dest	The relative path on Dropbox where the file should get uploaded.
overwrite	Default behavior (TRUE) is to overwrite files in the destination. Set to FALSE to prevent this.
autorename	This value, either true (default) or false, determines what happens when there is a conflict. If true, the file being uploaded will be automatically renamed to avoid the conflict. (For example, test.txt might be automatically renamed to test (1).txt.) The new name can be obtained from the returned metadata. If false, the call will fail with a 409 (Conflict). response code.
verbose	By default verbose output is FALSE. Set to TRUE if you need to troubleshoot any output or grab additional parameters.
encode	The file encoding
dtoken	The Dropbox token generated by drop_auth . rdrop2 will try to automatically locate your local credential cache and use them. However, if the credentials are not found, the function will initiate a new authentication request. You can override this by pointing to a different location where your credentials are stored.

Examples

```
## Not run:  
write.csv(mtcars, file = "mtt.csv")  
drop_upload("mtt.csv")  
  
## End(Not run)
```

Index

drop_acc, 2
drop_auth, 2, 3, 4–15
drop_copy, 3
drop_create, 4
drop_delete, 5
drop_delta, 6
drop_dir, 6
drop_exists, 8
drop_get, 9
drop_history, 10
drop_media, 10
drop_move, 11
drop_read_csv, 12
drop_search, 13
drop_share, 14
drop_upload, 14